Communication Protocols
and Internet Architectures

Harvard University
CSCI S-1

Lectures #9 and #10

Instructor: Len Evenchik
(evenchik@fas.harvard.edu)

© 1998-2005 L. Evenchik

Lecture Agenda

• Logistics and Q & A
• Application Layer Protocols
• Network Security
• Network Security
  – In addition to the technical issues, there are many legal issues
    which must be considered when designing and implementing
    security systems. This material DOES NOT discuss these
    issues at all. Before you implement a security solution, talk to
    a knowledgeable attorney regarding company and personal
    liability, and other legal issues.
• Public Key Encryption
• Hash Functions and Digital Signatures
• VPNs and IPSec
• Three Minute Wrap-Up

© 1998-2005 L. Evenchik
Course Logistics

- The graded midterms will be handed back tonight. The mean and median on the exam was 44 points.
- The final exam will be held next Wednesday night August 17, 2005 in Science Center lecture hall D starting at 6pm. July 27, 2005.
- Homeworks #4 and #5 are being distributed tonight and are due next week.

Network Security

© 1998-2005 L. Evenchik
Security Resources
No Single Resource is Enough

• CERT Coordination Center
  – 412-268-7090
  – www.cert.org
• Your corporate IT group and legal department.
• IETF working groups
• Your ISP
• Your firewall, router and other equipment vendors
Where Should Security be Used in this Network?

Security - Some of the Threats and Problems

- Wiretapping
- Breaking into computer systems
- Stealing information
- Denial of service
- Replay of prior conversations
- Masquerading as someone else
- Changing the message
Security - Some Current Building Blocks

- Physical security for systems and networks
- Password security for systems and networks
- Shared secret encryption system
- Public key encryption system
- Routers and firewalls
- IPsec and SSL/TLS
- Digital signatures
- Proper procedures

Security is a system issue which requires hardware, software, training and procedures.

© 1998-2005 L. Evenchik

Security – Functional Areas

- Privacy and confidentiality
- Authentication
- Authorization
- Integrity
- Nonrepudiation

© 1998-2005 L. Evenchik
Cryptography

Approaches to Cryptography

• Symmetric cryptography
  – Shared secret key system
  – Same key used to encrypt and decrypt messages
  – Key length determines the “strength” of encryption
  – Key management is difficult
  – Examples are DES, 3DES, IDEA, RC4 and AES

• Asymmetric cryptography
  – Key pair - one public, one private
  – Data encrypted by one key and decrypted by the other
  – Examples are Diffie-Hellman (1976) and RSA (1978)
Data Encryption Standard (DES)

- DES is a shared secret key encryption scheme. It is FIPS standard 46-3 and it is now over 25 years old.
- Encryption in 64 bit blocks with chaining or feedback
- Key length determines the “strength” of encryption.
- DES uses a 56 bit key; this is now considered weak encryption and DES is no longer endorsed for US Government use.
- Triple DES (3DES) can be used in place of DES

Advanced Encryption Standard (AES)

- Advanced Encryption Standard (AES) is a shared secret key encryption scheme; it is the successor to DES
- Developed by NIST using a public evaluation process (15 candidates.) AES published in Nov. 2002.
- AES is a symmetric block cipher encryption algorithm
- AES supports key lengths of 128, 192 and 256 bits.
Public Key Algorithms

- Key distribution and management has always been the weak link in shared key systems
- Public key systems solve this problem by being able to publish a “public key”
- Algorithm must provide the following functionality:
  - \( D(E(P)) = P \)
  - It is very difficult to deduce \( D \) from \( E \)
  - \( E \) cannot be broken by a chosen plaintext attack
- Appropriate algorithms are based on hard problems such as taking the log of a number or factoring large numbers.

ssh-keygen

```
host@dcepea (~): ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/web/c/s/cscie131b/.ssh/id_rsa): test4
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in test4.
Your public key has been saved in test4.pub.
```

© 1998-2005 L. Evenchik
Private Key

```
cmd (~): cat test4
-----BEGIN RSA PRIVATE KEY-----
MIICXQIBAAKBgQC/aSKmm6VdcqL6IqzK81998Ac8coes/V214KGEItcSYboSE1e7
s3RVssdY9Xqol1cVEXhQq/SznQhKxi4CrC6dxyOwpVdDSD07ZW8LRg2Gw1jFOU
KU0LeEsqzmEBdtewuixbDITAeMQjtRnjdPo8f1i37MW5sS2ZvdpwesKwWIBiWK
qGC58hp1hTTHLHygmsQy3cp6VuJ01lwd02N6hi1snC+beqBFXC3nMR+1akGnhY53
433q6A44V76x+MY58s2cmtj9/5Lleofpt/uW53056TZwniPgAQXiqe5nPpwdNZ
ve0oVOpHUYFYjvYTAAZCm68baY16FpESYdp0wYdOlawJBAOjuhwyNx3u47po
VGJnfhNNs0s8wzaA4U26heTVE8nxqRwIoVsn4REJNDByXqJ6pyLJHEVq6Y91JnJ
QnwYmCQoD80FugasGuZsgKQyg56ngdKXqg0y595LFcda+8krVuPdBnE6S1h
Iappr/6YRMVtVc0vaFUyTaAaDK1LaAkJfEAI3Drw1m0k5GI1FotFlk/BWQMFg8K
nFlTeqKqpvVUzaC1C182Ns2akzusoZs73b/scd5NNDE9Xo6qdyUjql+iwJBAAln
Kv95ycj969q0Y998Hg1Xkdwv1l57Hsv/JrFx2/fopF4lJbxAJUxp69B7pmrz
6+PnTe9K9WqCpq21UCQ6DkdkUUmcmTuF1szKX/M+5sMwdfpEvrt7FICBeQ14X8
iak2TSioziUBQy4Yuf38oCJX+QVESdi8P0vTVdUIi3
-----END RSA PRIVATE KEY-----
cmd (~): 
```

Public Key

```
cmd (~): cat test4.pub

ssh-rsa AAAAB3NzaC1yc2EAAAABIAwAAAIAE
Av2kippulXXKi+iEMyvNfffAHPAqrHflpheCh
mSLXEmG6EhNXu7N0vBLHWPVqJ3dXFRF4
YUPp83ISrYuAqwunnccjzKVQ3Q0q2VWC1
kYNhsNYxaFCq1lHi7BKm85hAxbXrr4aW1CE2
jBqo7Sp43RaFH4t2OzFubEmtb3acHkpF=
cmd (~): 
```

© 1998-2005 L. Evenchik
http://pgp.mit.edu/

MIT PGP Public Key Server

Key Server Status: Running normally.
Help: Generating keys / Submitting keys / Email interface / About this server / FAQ
Related Info: Information about PGP / MIT disponible site for PGP

Extract a key

Search String

Do the search

Index: O Verbose Index: C
○ Show PGP signposts for keys
○ Only return exact matches

Submit a key

Enter ASCII-armor4 PGP key here:

---

Search for pgp keys for “Harvard”

Public Key Server -- Index `"harvard "`

Type hit/KeyID Date User ID
pub 10243/2132841 1999/12/31 Peter Bonney <bonney@fas.harvard.edu>
pub 10243/88312640 2003/12/02 Clayton Carter <carter@cs.indiana.edu>
        Clayton Carter (new) <carter@cs.indiana.edu>
pub 10243/88312640 2003/11/13 Alberto Acosta (adapt key) <alberto@cs.indiana.edu>
pub 10243/46725870 2003/10/10 TS Thom <timesmith@fas.harvard.edu>
pub 10243/11123444 2003/10/10 Edward Kacmarcik <kacmarcik@fas.harvard.edu>
pub 10243/11123444 2003/10/15 Alex Casey <casey@fas.harvard.edu>
pub 10243/31025211 2003/09/26 Giles Ho <giles@fas.harvard.edu>
pub 10243/7775716 2003/09/07 M. Brandon Swain <mswain@fas.harvard.edu>
pub 10243/7775716 2003/09/07 M. Brandon Swain <mswain@fas.harvard.edu>
pub 10243/5c85e54 2003/09/05 James Maguire <jmg@fas.harvard.edu>
pub 10243/1e198114 2003/08/11 Richard Ryder <rjryder@fas.harvard.edu>
pub 10243/70e12c40 2003/07/02 Mark F. Kornbluth <mark@fas.harvard.edu>
pub 10243/4b809c50 2003/07/08 Paul Ksiez <pksiez@fas.harvard.edu>
pub 10243/3f818b2c 2003/05/24 James McIntosh <jm@mit.edu>
        James McIntosh <james McIntosh@mit.edu>
pub 10243/1093f4f2 2003/05/05 Mike Madison <madison@fas.harvard.edu>
pub 10243/977c5e00 2003/05/26 Evan R. Mohr <emohr@fas.harvard.edu>
pub 10243/1e198114 2003/05/14 James Maguire <jmg@fas.harvard.edu>
        Floriom Forestman <-fforestman@fas.harvard.edu>
        Floriom Forestman <-fforestman@fas.harvard.edu>
pub 10243/956c0ed0 2003/05/14 Floriom Forestman <-fforestman@fas.harvard.edu>
        Floriom Forestman <-fforestman@fas.harvard.edu>

(C) 2005 L. Evenchik
Search for pgp keys for “CERT”

Public Key Server -- Index ``cert``

Portion of the CERT key

Public Key Server -- Get `"0xFF755015"`
http://pgp.mit.edu/ - submit a key

MIT PGP Public Key Server

Key Server Status: Running normally
Help: Extracting keys / Submitting keys / Speak interface / About this server / FAQ
Related Info: Information about PGP / MIT distribution site for PGP

Extract a key

Search string:
Search Index: 0

Submit a key

Enter ASCII-armored PGP key here:

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABAAABHkKCI7SsJwSWf3dH4Ay5dFgQi93l3jFy8z
b3ME54p9Pm1TtJHct7P05FMC3I9F2p6wM3w3

http://www.cert.org/contact_cert/encryptmail.html

Sending Sensitive Information to the CERT®/CC

We strongly encourage you to store sensitive information. We can exchange email with your email PGP at CERT. We have PGP and PGP Public Keys, and a secure means of exchange, all at the click of a button.

We also encourage you to check out PGP's signature on email and documents.

PGP

As a good security practice, users should be sure to validate PGP keys they receive and extract unverifiable keys. In the past, forged CERT/CC PGP keys have been created and uploaded to the public key servers. It is important to validate the copy of the CERT/CC PGP public key to make it legitimate.

1. Get your PGP public key from the CERT/CC website or from a PGP Public Key Server such as the one at Massachusetts Institute of Technology.

The new key has a RSA key, and it is intended to provide maximum interoperability with as many versions of PGP as possible, as well as with other systems.

Information about the old PGP key can be found here.
Public Key Algorithms (Part 2)

Public key algorithms are much slower than symmetric key algorithms. Therefore a combination of the two approaches is used by many systems:

• Sender does….
• Recipient does….

Let’s fill in the details.....

Very Basic Security Issues
Basic Computer Security

- Host (computer) based approach provides marginal security in a networked environment.
- Classic and most common security is based solely on password protection and passwords are not difficult to break. Multi-factor security is better.
- Consumer and business PCs are shipped with “friendly” password settings and provide no security protection.
- Important to provide physical security to prevent brute force methods of attack. Being able to reboot a system gives you all sorts of options.

Wiretapping and Snooping

- It is not difficult to physically wiretap, or monitor the traffic on a network as it is sent.
- Straightforward way to capture passwords and other sensitive information. Hence the need for encryption.
- Very difficult to prevent it or detect that it is going on
- One way to reduce the risk (but only by a marginal amount) in a network environment is to segment the network traffic using ethernet switches and routers. Why does this help?
Link Encryption

- Provides encryption on a per link point to point basis
- Prevents compromise of information from wiretapping and snooping
- Can be used in combination with other techniques
- Simple implementation but difficult management in large systems due to key management
- Mature, relatively inexpensive approach

Security Provided by Routers and Firewalls
Router Based Security

- Router access control lists (ACL) can filter packets based on one or more criteria:
  - Source and/or destination IP address
  - Transport layer protocol (UDP vs TCP)
  - Application protocol (H.323, FTP, HTTP, DNS, etc.)
  - Plus others
- Router access lists are not easy to maintain

Basic Firewall Architecture

- Firewalls are typically dedicated network devices that isolate the external network from the internal/enterprise network.
- Firewalls are implemented using custom hardware and software, or software running on a UNIX, LINUX or Windows platform.
- Firewalls should be managed separately from the enterprise routers
Firewall Functionality (part one)

- Firewalls implement a security policy for the enterprise based on a defined set of access rules.
- Rules define what is allowed into the enterprise network from the outside world as well as what is allowed out of the network.
- Firewalls can implement access security using a number of different technical approaches (all of them have been confused by the vendors):
  - Packet based filter/forward decisions
  - Stateful Inspection
  - Application Layer Gateways

Firewall Functionality (part two)

- Firewalls decisions can be based on:
  - Source and/or destination address
  - Both transport and Application layer protocols (TELNET, FTP, HTTP, etc)
  - The specific user
  - Application layer URLs and MIME types
  - Differences between inbound versus outbound traffic
- Firewalls can also provide user authentication:
  - Simple user passwords or one time passwords
  - This functionality should be in addition to the filtering functionality noted above
Firewall Functionality (part three)

- ALGs terminate application layer sessions for individual users and then create new ones depending upon security policy.
- Firewalls can provide NAT and service redirection
- Firewalls can provide secure tunnels with encryption for remote users (aka VPNs)
- Firewalls should provide extensive logging, reporting, management and alarms
- Firewalls features are being added constantly. For example: virus checking, spam filters, QoS, etc.
- Firewalls are not a complete security solution
Firewalls and Proxies are Different

Firewall Architecture - option 1
Hashing, Message Digests and Digital Signatures

Hashing Functions and Message Digests

• Hash functions take an arbitrarily long piece of plaintext and compute from it a fixed length string.

• Hash functions are based on the fact that there are transformations that are easy to do but very, very hard to undo.
  – In mathematical terms $y=f(x)$
  – Given $f$ and $x$, it is very easy to compute $y$
  – Given $f$ and $y$, it is very hard to compute $x$

• Message digests should be 128 bits or longer

• MD5 is the 5th hash function designed by Ron Rivest (1992)

• Hash functions do not provide confidentiality.

• What is the Birthday Attack?
Print of “testfile1”

```
cmd (~): cat testfile1
this is a test file to be used in the networks and protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World
This is line five (5) of this file.
cmd (~):
```

MD5 of a file called “testfile1”

```
cmd (~): cat testfile1
this is a test file to be used in the networks and protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World
This is line five (5) of this file.
cmd (~):

cmd (~): md5 testfile1
MD5 (testfile1) = 92af22d896cb2e8583ba7deb74b4e614
cmd (~):
```

© 1998-2005 L. Evenchik
Comparison of “testfile1” and “testfile2”
Note the small difference on the line with “Hello World”

```
cmd (~): cat testfile1
this is a test file to be used in the networks and protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World
This is line five (5) of this file.
cmd (~):
```

```
cmd (~): cat testfile2
this is a test file to be used in the networks and protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World!
This is line five (5) of this file.
cmd (~):
```

MD5 Comparison for files “testfile1” and “testfile2”

```
cmd (~): md5 testfile1
MD5 (testfile1) = 92af22d896cb2e8583ba7deb74b4e614
cmd (~):
```

```
cmd (~): md5 testfile2
MD5 (testfile2) = 8a09807d72d18892718e1c157b54ab7f
cmd (~):
```
Digital Signatures

• A digital signature should “prove” that a message came from a specific user (lets call that user, UserA)

• One way to produce a digital signature
  – Step 1… UserA computes a ……..
  – Step 2…..

Signing of “testfile1”

cmd (~): gpg --clearsign < testfile1

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

this is a test file to be used in the networks and protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World
This is line five (5) of this file.
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.0.7 (SunOS)
iQCVAwUBP8u0S+nwDzqNmKQTAQIzEAQAiYTHo
PS4GZMUjFyztigG2nWXu13867oYvvPp/D9q+jTR6O
PapnwowXpggIlZn0mluxMoTO0pSkjgcC3ILqo0o4
W5z6B8iykfdXoyDMCu4+n133OgjjYS/lyIrq9org+
gEw9nn4ChyyqSLvbHwgo1B6fr1ml+HGhP4PvwDCM=
=ZQMq
-----END PGP SIGNATURE-----
cmd (~):
cat of signature file of “testfile1sign”

```bash
cmd (~): cat testfile1sign
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

this is a test file to be used in the networks and
protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World
This is line five (5) of this file.
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.0.7 (SunOS)
iQCVAwUBP8u0S+nwDzqNmKQTAQIzEAQAIYTHo
PS4GZMUjFyztitigG2nWXul3867oYuvPp/D9q+jTR6O
PapnwowXpgqJIZn0mluxMoTO0p5kygC3ILqo04
W5z6BN8ykfdXoyDMCuh4+n1330qjjYS/llyrq9org+
gEw9nn4Chyyq5LvbHwgo1B6fr1ml+HGi4P4PvwdCDM=
=ZQMq
-----END PGP SIGNATURE-----
cmd (~):
```

Signature Verification

```bash
cmd (~): gpg --verify testfile1sign
```
gpg: Warning: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
gpg: Signature made Mon Dec 01 16:36:11 2003 EST
using RSA key ID 8D98A413
gpg: Good signature from "course test (test key for demo)
    <cscie131b@barkley.dce.harvard.edu>"
cmd (~):
```
cat of signature file of “testfile1sign”

```
cmd (~): cat testfile1sign
-------BEGIN PGP SIGNED MESSAGE-------
Hash: SHA1

this is a test file to be used in the networks and protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World
This is line five (5) of this file.
-------BEGIN PGP SIGNATURE-------
Version: GnuPG v1.0.7 (SunOS)
iQCVAwUBP8u0S+nwDzqNmKQTAQlzEQAQiYTHo
PS4GZMuJFyztligG2nWXuI3867oYywPp/D9q+jTR60
PapnwowXpqqIJzn0mluxMoTO0pSkgycC3lLqo0o4
W5z6BN8ykfdXoyDMCuh4+n133OgjjYS/lyrq9org+
gEw9nn4Chyyq5LvbHwgo1B6fr1ml+HGI4P4PvwdCDM=
=ZQMq
-------END PGP SIGNATURE-------
cmd (~):
```

© 1998-2005 L. Evenchik

---

cat of signature file of changed “testfile1”

Note the added character at “Hello World”

```
cmd (~):
-------BEGIN PGP SIGNED MESSAGE-------
Hash: SHA1

this is a test file to be used in the networks and protocols class...
abcdefghijklmnopqrstuvwxyz1234567890
Hello World ?
This is line five (5) of this file.
-------BEGIN PGP SIGNATURE-------
Version: GnuPG v1.0.7 (SunOS)
iQCVAwUBP8u0S+nwDzqNmKQTAQlzEQAQiYTHo
PS4GZMuJFyztligG2nWXuI3867oYywPp/D9q+jTR60
PapnwowXpqqIJzn0mluxMoTO0pSkgycC3lLqo0o4
W5z6BN8ykfdXoyDMCuh4+n133OgjjYS/lyrq9org+
gEw9nn4Chyyq5LvbHwgo1B6fr1ml+HGI4P4PvwdCDM=
=ZQMq
-------END PGP SIGNATURE-------
cmd (~):
```

© 1998-2005 L. Evenchik

---
Failure of Signature Verification

cmd (~):
gpg --verify testfile1badsign

```
gpg: Warning: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
gpg: Signature made Mon Dec 01 16:36:11 2003 EST
    using RSA key ID 8D98A413

gpg: BAD signature from "course test (test key for demo)
    <cscie131b@barkley.dce.harvard.edu)"
```

cmd (~):

Digital Signatures

- A digital signature should “prove” that a message came from a specific user (let’s call that user, UserA)
- One way to produce a digital signature
  - UserA computes a one-way hash function on the contents of the message
  - UserA encrypts the hash code using its private key
  - The encrypted hash code is appended to the message and the combination is sent to UserB
  - UserB computes the same hash function on the contents of the message
  - UserB then decrypts the received hash code with UserA’s public key
  - If the hash codes match, the message came from UserA and the message was not changed in transit
SSL and Certificates

Providing Secure Web Access via SSL

Secure Data Access over the Internet

There are 4 basic components of information security to keep in mind when accessing online applications like PIN.

1. The PIN itself
2. Data on the Computer You Are Using
3. Data in Transit over the Internet
4. Data Stored on a Secure Computer

1. The PIN itself

Keeping your PIN confidential is essential to ensure the privacy of University information that you store. The following guidelines to protect the privacy of your PIN:

• Never share your PIN with anyone, even if they ask for it, including University staff, colleagues.
• If you received your initial PIN in a printed letter, keep it in a secure location until you change it.
Browser Certificate Management

Certificate Information
Virtual Private Networks and IPsec
Virtual Private Network

- Sites can be connected together via the public internet or private network facilities (circuit)
- Tunnel mode provides a secure encrypted tunnel between sites. Note that this is not the same as supporting individual users via Transport mode.
- Can be implemented as part of a firewall or as a stand-alone product.
- One access circuit can provide both VPN service and public access to the Internet at the same time.
- Individual clients are supported by software for access via dial-up, cable or xDSL.
- Multiple tunneling protocols are available, both proprietary and public (IPSec)

VPN Architecture
**IPsec**

- Developed by working group of the IETF
- Provides confidentiality, integrity and authentication for IP packets
- Provides for encapsulation of protected information
- Authentication Header (AH) - cryptographic checksum of contents
- Encapsulated Security Protocol (ESP) - provides confidentiality of contents and can provide authentication
- Both Tunnel and Transport Mode supported
- Key management and exchange is separate
- IPv6 requires that IPSec be implemented
- See RFCs 2401 and 2411 as your starting point.

---

**AH Header**

**TRANSPORT MODE**
AH is inserted after IP header and before any upper layer protocol

**BEFORE APPLYING AH**

```
-----------------------------
IPv4 | orig IP hdr | (any options) | TCP | Data |
-----------------------------
```

**AFTER APPLYING AH**

```
---------------------------------
IPv4 | orig IP hdr | AH | TCP | Data |
---------------------------------
```

except for mutable fields

**TUNNEL MODE**
Use of AH in either hosts or security gateways

```
-----------------------------
IPv4 | new IP hdr* | orig IP hdr* | (any options) | AH | (any options) | TCP | Data |
-----------------------------
```

<- authenticated except for mutable fields -->
in the new IP hdr

Source RFC 2402
ESP Header

TRANSPORT MODE
ESP is inserted after IP header and before any upper layer protocol

BEFORE APPLYING ESP

IPv4 | orig IP hdr | | | | (any options) | TCP | Data |

AFTER APPLYING ESP

IPv4 | orig IP hdr | ESP | | ESP | ESP | (any options) | Hdr | TCP | Data | Trailer | Auth |

TUNNEL MODE
Use of AH in either hosts or security gateways

IPv4 | new IP hdr* | orig IP hdr* | | | | ESP | ESP | (any options) | Hdr | TCP | Data | Trailer | Auth |

Source RFC 2406
© 1998-2005 L. Evenchik

Denial of Service

© 1998-2005 L. Evenchik
Denial of Service Attacks

- Generic problem of preventing authorized users from being able to use a system or resource.
- Attacks can be straightforward or very technically sophisticated
- Some classic attacks noted in the press:
  - computer and network viruses
  - email flooding and web server flooding
  - TCP SYN flooding
  - Oversized PING packets (one of the first DOS attacks…)
- Distributed Denial of Service attacks are the even more difficult to defend against.
- It is critical to keep your system up to spec with software patches to help guard against known attacks. (Again, this is only part of the solution.)

Three Minute Wrap-Up

- Please write down the three or four major points that were discussed during the lecture.
- Note whether the material was presented clearly and/or how it should have been done differently.
- Ask any questions that I should address next time.
- Please do a Wrap-Up at the end of lecture and hand it in as you leave or fill out the form on the web site.
- Do not sign your form. (The form on the web site is also anonymous.)
- Thank you!