Minicase studies on real-life COBIT® applications that can be used as small exercises in undergraduate and graduate classes
The IT Governance Institute (ITGI) was established in 1998 to advance international thinking and standards in directing and controlling an enterprise’s information technology. Effective IT governance helps ensure that IT supports business goals, optimises business investment in IT, and appropriately manages IT-related risks and opportunities. The IT Governance Institute offers symposia, original research and case studies to assist enterprise leaders and boards of directors in their IT governance responsibilities.

With more than 35,000 members in more than 100 countries, the Information Systems Audit and Control Association (ISACA) is a recognised worldwide leader in IT governance, control, security and assurance. Founded in 1969, ISACA sponsors international conferences, publishes the Information Systems Control Journal, develops international information systems auditing and control standards, and administers the globally respected Certified Information Systems Auditor (CISA) designation, earned by more than 35,000 professionals since inception, and the Certified Information Security Manager (CISM) designation, a groundbreaking credential earned by 5,000 professionals in its first two years.

The IT Governance Institute, Information Systems Audit and Control Association [the “Owner(s)’] and the authors have designed and created CobiT in Academia and its related publications, titled CobiT® Caselets, CobiT® Student Book, CobiT® Case Study: TIBO, and CobiT® Presentation Package, (the “Work”), primarily as an educational resource for educators. The Owners make no claim that use of any of the Work will assure a successful outcome. The Work should not be considered inclusive of any proper information, procedures and tests or exclusive of other information, procedures and tests that are reasonably directed to obtaining the same results. In determining the propriety of any specific information, procedure or test, the educator should apply his/her own professional judgement to the specific circumstances presented by the particular systems or information technology environment.

Copyright © 2004 IT Governance Institute. All rights reserved. This publication is intended solely for academic use and shall not be used in any other manner (including for any commercial purpose). Reproductions of selections of this publication are permitted solely for the use described above and must include the following copyright notice and acknowledgement: “Copyright © 2004 IT Governance Institute. All rights reserved. Reprinted by permission.” CobiT in Academia may not otherwise be used, copied, or reproduced, in any form by any means (electronic, mechanical, photocopying, recording or otherwise), without the prior written permission of the IT Governance Institute. Any modification, distribution, performance, display, transmission, or storage, in any form by any means (electronic, mechanical, photocopying, recording or otherwise) of CobiT in Academia is strictly prohibited. No other right or permission is granted with respect to this work.

IT Governance Institute
3701 Algonquin Road, Suite 1010
Rolling Meadows, IL 60008 USA
Phone: +1.847.590.7491
Fax: +1.847.253.1443
E-mail: research@isaca.org
Web sites: www.itgi.org and www.isaca.org

ISBN 1-893209-96-2
CobiT in Academia
Printed in the United States of America
IT GOVERNANCE INSTITUTE WISHES TO RECOGNISE:

The Board of Trustees
Marios Damianides, CISA, CISM, CA, CPA, Ernst & Young LLP, USA, International President
Abdul Hamid Bin Abdullah, CISA, CPA, Auditor General’s Office, Singapore, Vice President
William C. Boni, CISM, Motorola, USA, Vice President
Ricardo Bria, CISA, SAFE Consulting Group, Spain, Vice President
Everett C. Johnson, CPA, Deloitte & Touche LLP (retired), USA, Vice President
Howard Nicholson, CISA, CRN Solutions, Australia, Vice President
Bent Poulsen, CISA, CISM, VP Securities Services, Denmark, Vice President
Frank Yam, CISA, CIA, CCP, CFE, Focus Strategic Group Inc., Hong Kong, Vice President
Robert S. Roussey, CPA, University of Southern California, USA, Past International President
Paul A. Williams, FCA, MBCS, Paul Williams Consulting, UK, Past International President
Emil D’Angelo, CISA, CISM, Bank of Tokyo-Mitsubishi, USA, Trustee
Ronald Saull, CSP, Great-West Life and IGM Financial, Canada, Trustee
Erik Guldentops, CISA, CISM, Belgium, Advisor, IT Governance Institute

The Development Team
Erik Guldentops, CISA, CISM, University of Antwerp Management School, Belgium (Chair)
Roger Debreceny, Ph.D., FCPA, University of Hawaii, USA
Steven De Haes, University of Antwerp Management School, Belgium (Project Manager)
Roger Lux, Farmers Insurance Group, USA
John Mitchell, CISA, CIA, CFE, LHS Business Control, UK
Ed O’Donnell, Ph.D., Arizona State University, USA
Scott Summers, Ph.D., Brigham Young University, USA
Wim Van Grembergen, Ph.D., University of Antwerp Management School, Belgium

Expert Review
José Carrillo Verdún, Ph.D., Facultad de Informática, Universidad Politécnica de Madrid, Spain
TABLE OF CONTENTS

Purpose of This Document.................................................................................................................. 5
Caselet: Union National Bank............................................................................................................. 6
  Case Description.............................................................................................................................. 6
  Teaching Notes.............................................................................................................................. 6
    Extra Information......................................................................................................................... 6
    Questions.................................................................................................................................... 7
    Suggested Solutions..................................................................................................................... 7
Caselet: Lux Insurance Brokers........................................................................................................... 8
  Case Description.............................................................................................................................. 8
  Teaching Notes.............................................................................................................................. 9
    Case Background......................................................................................................................... 9
    Extra Information....................................................................................................................... 10
    Questions.................................................................................................................................... 12
    Suggested Solutions..................................................................................................................... 12
Caselet: Phoenix Electric Provider.................................................................................................. 17
  Case Description.............................................................................................................................. 17
  Teaching Notes.............................................................................................................................. 18
    Extra Information....................................................................................................................... 18
    Questions.................................................................................................................................... 19
    Suggested Solutions..................................................................................................................... 20
**PURPOSE OF THIS DOCUMENT**

CobIT Caselets is a product of the IT Governance Institute (www.itgi.org), and part of CobIT in Academia. This document was developed in collaboration with a group of international academics and practitioners. These caselets provide three minicase studies on CobIT applications in real-life situations and are designed for use as small exercises (90-minute class period) in undergraduate and graduate classes on information systems management, information security management, auditing, information systems auditing and/or accounting information systems. The objectives of these cases are to:

- Provide a foundation for students to identify risks and control areas in a rich information technology environment
- Provide a means for students to relate risks and controls to particular CobIT control objectives
- Help students learn how to use the CobIT audit guidelines to identify appropriate procedures when developing methodologies for providing assurance over controls

Each of the cases is composed of two parts. The first part is a case description, which provides background on the case organisation and its IT processes and infrastructure. Next, the teaching notes supply extra background information for the instructor, potential questions for the students, and suggestions on answers from the students.

There are three other components that make up CobIT in Academia. The CobIT Student Book explains and illustrates all the CobIT components. The CobIT Presentation Package provides a PowerPoint deck of 80 slides explaining all the CobIT elements. The CobIT Case Study: TIBO applies CobIT knowledge in a real-life situation.
CASE DESCRIPTION

Union National Bank (UNB) has history on its side. With well-established roots, the 149-year-old community bank has a reputation for old-fashioned customer service. However, it has recently found itself competing not only with other well-established local banks but also with giant conglomerates. The bank attempted to compete on price by lowering fees, discounting loan rates and inflating deposit rates, while still offering its hands-on customer service, but management discovered that was a money-losing move and decided to try another strategy.

During the current fiscal year, the seven-branch UNB embarked on a major change in its processes, products and pricing to stay profitable and still retain its customer-centric reputation by acquiring a customer relationship management (CRM) package. The CRM solution allows UNB to look more closely at not only its big-picture financials but also its profitability by customer. Management segmented the bank’s 37,000 accounts by profitability and drilled down into individual household information to get a view of spending and account activity. Management then fed all bank transactions into a single data warehouse, where they could be analysed later to pinpoint the most important products to its clients. The software alerts member service representatives (assigned to individual accounts) to life changes that may indicate an interest in another financial product. Recently married customers may be interested in mortgage services. Customers who are nearing 65 years of age may be interested in retirement products. The CRM solution also helps the bank remain on customers’ good side by asking how and when to contact them. This increases the likelihood of a positive response to a sales pitch. The bank has achieved more than US $1 million in efficiency gains, stemmed customer erosion and seen substantial revenue growth.

Management is using the CRM system tools to calculate the net profit of all its bank customers by analysing all associated costs and revenue associated with each customer’s behaviour. Did customers visit tellers often or opt for less-expensive ATM and Internet transactions? How many cheques were they writing a month? Was there other revenue associated with customer accounts from optional services or extra fees? By using this tool, the bank was able to segment its customer base into high-value and low-value buckets. When the bank began to raise some of its rates and fees to pay for these high-tech services, it was able to protect its most profitable customers from the increases. Bank officials also began to create incentives to make the less-valuable customer segments more profitable to the bank—either by encouraging electronic banking or signing customers up for products that generated more revenue.

The CRM solution enables management to figure out what new products it can safely introduce without competing with its existing offerings. For example, UNB recently used the system to help roll out its new savings account program. It hoped the year-end, high-minimum, short-term, high-interest offering would attract lucrative new customers with more than US $5,000 to invest. Management was able to use the CRM system to figure out how many accounts contained more than the minimum balance, the difference in rates of interest, and the potential financial loss if some or all of them switched to the new, higher rate account. Management could then balance that potential loss against potential new revenue.

TEACHING NOTES

EXTRA INFORMATION

This case was excerpted from an article by Stephanie Overby, “The Little Banks that Could,” which was published in the 1 June 2002 edition of CIO Magazine. The design of the case is to illustrate the processes that
an organisation uses to align information technology solutions with strategic business objectives, manage the implementation of those solutions and monitor how well the solution meets performance objectives after systems are implemented.

QUESTIONS
You are the external auditor for UNB.

1. Employ the PO1 define a strategic IT plan CobiT process to:
   a. Identify which of the PO1 control objectives are most appropriate for you to consider in designing your audit plan. Be prepared to justify your selection of the relevant control objectives.
   b. For three of the control objectives that you chose in (a), employ the CobiT audit guideline for PO1 to develop a draft set of audit procedures to provide assurance on the achievement of these control objectives.

2. Employ the PO10 manage projects CobiT process to:
   a. Identify which of the PO10 control objectives are most appropriate for you to consider in designing your audit plan. Be prepared to justify your selection of the relevant control objectives.
   b. For three of the control objectives that you chose in (a), employ the CobiT audit guideline for PO10 to develop a draft set of audit procedures to provide assurance on the achievement of these control objectives.

3. Employ the M1 monitor the process CobiT process to:
   a. Identify which of the M1 control objectives are most appropriate for you to consider in designing your audit plan. Be prepared to justify your selection of the relevant control objectives.
   b. For three of the control objectives that you chose in (a), employ the CobiT audit guideline for M1 to develop a draft set of audit procedures to provide assurance on the achievement these control objectives.

SUGGESTED SOLUTIONS
The instructor should consider the following with regard to the three case questions:

1. Defining a strategic IT plan has become central to UNB’s strategy for achieving competitive advantage because the new strategy depends heavily on customer service processes that require IT support. As a result, establishing controls that ensure adequate planning for linking IT performance objectives with the new strategic businesses is of primary importance (control objectives PO1.1 through 1.5).

2. Managing the software implementation project effectively is central to achieving a cost-effective CRM solution. The bank’s goal was to gather and act on customer information efficiently enough to make the process profitable. The cost of the implementation is the primary driver of the efficiency of this type of solution. Control objectives within the PO domain increase the likelihood that (10.1) the project is well defined, (10.3) the right personnel are involved in the implementation, (10.7) the master plan is adequate for maintaining control, (10.11) the system is adequately tested, (10.12) personnel are properly trained to use the system and (10.13) a thorough post-implementation review is conducted to identify which potential problems are the most important.

3. Upon implementation of the solution, a process for monitoring its effectiveness must be put in place. Monitoring involves identifying how the solution will help the organisation achieve its business objectives, specifying IT performance goals that support the business objectives, developing key performance indicators (KPIs) that allow management to monitor whether the system is meeting those performance objectives, and then creating a process for generating KPI data and making them available to management on a timely basis. Helping the organisation achieve these goals is the focus of control objectives M1.1 through 1.4.
CASELET: LUX INSURANCE BROKERS

CASE DESCRIPTION

Lux Insurance Brokers (LIB) is an insurance brokerage firm based in Sydney, New South Wales, Australia. LIB provides house, car and life insurance to individuals residing in Australia. In the last couple of years, LIB has developed an online business model to complement its storefronts. LIB has a different online business model from the other insurance brokers who maintain an online presence. LIB’s competitors search out the best insurance pricing for their clients. While the broker facilitates the relationship, the final transaction is between the client and the insurance company directly. The broker subsequently receives a commission from the insurance company. Conversely, LIB provides a one-stop service for its clients, from search to completion of the contract. LIB charges a fee to the client and takes no commission from the insurance company. Under a new Australian law, LIB is able to produce the insurance contract and send it electronically to the client and insurance company, without requiring that the client sign it. The director of IT heads the IT functions at LIB. Reporting to the director are the managers of the program development, network operations and operations groups.

Pricing of insurance products for individuals is highly dependent on a small number of factors, including creditworthiness, claim history and the nature of risk insured. Approximately 70 percent of the online business conducted by LIB is vehicle insurance. LIB validates the client’s credit card number against credit history, vehicle identification number (VIN) against the Department of Motor Vehicles (DMV) registration database, and client information against the shared database of policy claim and payment history maintained by the Insurance Industry Information Council (IIIC). All of this information is collected from the client on the Internet. Credit card payment for the policy comes from the client after final processing of the policy.

The client interacts with a web-based system. The client registers on the web site and provides all requested personal and insurance information. The system uses a complex set of JavaScript codes to keep the transaction alive, as the client clicks from page to page and interacts with LIB’s own back-end systems as well as the systems of the DMV, IIIC, etc. LIB’s databases and temporary files in the LIB file systems cache some of the data transmitted to and from these third parties. Some data pass directly through to the third party without caching. The web site provides no information on the handling of the data for the clients. The client can allow an automatic logon to the site. The browser’s cookie file stores the challenge information (username and password). When clients forget their username and/or password, they can answer a challenge question to retrieve the information by e-mail.

The JavaScript codes, used by LIB for its web interaction with clients, are derived from a stock suite of e-commerce code that is sold by a local Australian e-commerce software developer. The e-commerce code is self-documenting. This means the body of the code incorporates all the help files and guidance for the integrator. The software developer finds this more efficient than providing a series of manuals. Most clients do not use all the available code in the suite and alter the code to respond to the particular development environment. Indeed, the development team in the LIB program development IT group has made a series of code changes to the JavaScript to increase performance and meet particular business needs. These changes have also been self-documented.

A ColdFusion web management system manages interactions between the JavaScript code and the back-end and third-party systems. The LIB program development IT staff code the ColdFusion application. The network operations group also wrote and maintains some specialist code in the PHP language. The quality assurance team in the program development group is responsible for stress testing the online applications.
The operations group maintains a range of hardware platforms including mainframes and distributed systems (e.g., UNIX, Windows and Netware). The network operations group manages a range of network components (e.g., ISDN, routers and firewalls).

The operations manager, program development manager and network operations manager share responsibility for security at LIB. Each group maintains a formal security plan. The three managers, together with the director of IT and the director of internal audit, form a security task force that meets twice a year as the security management committee to co-ordinate security developments and plans. The network operations group includes a manager of network security, who is responsible for the security of internal and external networks.

Neither the operations manager nor the program development manager considers that his/her group is sufficiently large to warrant a full-time security person. The responsibility for security is 40 percent of the job responsibilities of the assistant manager of operations in the operations group and 30 percent of the job responsibilities of the quality manager in the program development group. While no formal security plan is prepared, each of the three groups is required to report to the biannual meeting of the security management committee. The report from each group should include a listing of security breaches and identified security vulnerabilities, but not every report includes such detail and sometimes the reports are somewhat sketchy.

Recently, the manager of network security in the network operations group has been reading about the risks that arise from cross-site scripting (XSS). This form of Internet vulnerability arises from a third party inserting malicious code in the clickstream between browser and server. While the malicious code can run on either the server or client side, most examples of XSS are on the client’s web browser. The fraudster exploits the intelligence of the web browser to hijack the browser and run JavaScript applets that may retrieve information from the client’s cookie file or send user data, including passwords and login information, to the third party. The security manager in the network operations group raises the possibility that LIB may be susceptible to XSS in his/her report to the security management committee. He/She also notes that it does not seem to be a vulnerability that affects the network operations group, as it does not influence the availability or vulnerability of LIB’s Internet connectivity, intranet or other proprietary networks. The committee noted the report.

TEACHING NOTES

CASE BACKGROUND
The foundation for this case is Charles Schwab & Co. Inc., a major user of COBIT. The base for the Charles Schwab business model is in large part the interaction of clients with their back-end systems via the Internet. Charles Schwab was the subject of an article on cross-site scripting vulnerability in 2000 [see the following article, “Cross-site Scripting (XSS) Vulnerabilities”]. This class of risk is current and important, given the complexity of web pages employed in e-commerce activities.

While aspects of the case might relate to a number of processes in COBIT, including DS2 manage third-party services and M1 monitor the processes, the primary relationship is to DS5 ensure systems security. The focus of the questions below is on this process. It is suggested that the control objectives, audit guidelines and management guidelines of COBIT process DS5 (as available in the COBIT Student Book, a part of COBIT in Academia) be provided to students. You may wish to distribute the following article on Charles Schwab and Co. Inc. (from the following list) after the case discussion.

\[1\] Troy Wolverton, Special to ZDNet, 6 December 2000 5:45 PM PT
EXTRA INFORMATION

In essence, XSS arises because of the complexity of embedded JavaScript codes in the clickstream between the e-commerce provider and the client web browser. A hacker exploits known vulnerabilities and this complexity to hijack the information flow and potentially the contents of the cookie file.

For the benefit of faculty, the following background materials provide information on cross-site scripting (XSS):

- Background article: “Cross-site Scripting (XSS) Vulnerabilities” (in the following paragraphs)
- IT Governance Institute, “Charles Schwab & Co., Inc. Implements COBIT and IT Governance,” www.isaca.org/Content/ContentGroups/CoBIT2/Case_Studies/Charles_Schwab.htm

Background Article: “Cross-site Scripting (XSS) Vulnerabilities”

Charles Schwab & Co.’s web site is vulnerable to a well-known attack that could allow a hacker to gain access to sensitive account information, the financial services company acknowledged Wednesday.

Reported by San Francisco-based programmer Jeff Baker on the Bugtraq security mailing list on Wednesday, the vulnerability involves “cross-site scripting.” The vulnerability, which uses popular web programming languages, such as JavaScript, to hijack a customer’s web browser, is similar to one acknowledged by E*Trade Group Inc. in September.

By exploiting the vulnerability, “malicious users can fool other users’ web clients...which allows them to do things such as stealing that client/server’s cookies,” Elias Levy, Bugtraq’s moderator and the chief technology officer of SecurityFocus.com, wrote in an advisory. Calling the vulnerability a “common flaw,” Levy blamed the problem in part on “the lack of good practices by programmers of web-based applications.”

Although security experts first issued a warning against cross-site scripting in February, security experts believe that dozens of sites may still be vulnerable to the attack.

Baker said he first notified Schwab of its vulnerability in late August. Although he exchanged several e-mails with Schwab about the problem, the company did not close the vulnerability, he said.

A fix is on the way

“The flaws still exist, and I have no reason to believe that they are in the process of being fixed,” Baker said in his advisory on Bugtraq. “Schwab should strive to fix problems when given (four)-month advance notice. They should raise their ethical standards to alert their paying customers whenever a system vulnerability is reported.”

Ibid.
But Schwab spokesman Greg Gable said the company has been working as quickly as possible to address the problem. After being notified of the vulnerability in August, Schwab took some minor steps to protect customers and plans to completely close the vulnerability by early next year via a computer change, he said.

“We take security issues extremely seriously,” Gable said. “We take aggressive steps to minimise the risk.”

But Gable played down the risk for customers, calling it a “very, very narrow possibility.” Gable said the delay in closing the vulnerability had to do with balancing the ease of use of the site and the need to test the fix before implementing it.

“With a large system and a large customer base, we need to test thoroughly,” he said.

Cross-site scripting allows hackers to run dangerous code within [an Internet] user’s browser or e-mail client. Basically an attack on a Schwab user could allow the hacker to have access to all of the customer’s account actions—such as buying and selling stocks or transferring funds while the customer was logged on to his account.

To protect against such an attack, Baker warned Schwab customers to disable JavaScript in their browsers, avoid accessing other web sites or opening e-mail while logged on to their Schwab accounts, and completely log off Schwab’s site and close browsers when finished accessing accounts.

Background Article: “What Can an XSS Attack Do?”

On first glance, it might seem that cross-site scripting (XSS) attacks would not allow attackers to do much more than make nuisances of themselves. Unlike SQL poisoning, a scripting attack cannot alter or destroy an application’s back-end data, much less issue an arbitrary system call. JavaScript is a well-known and largely benign technology; web designers and developers use it on a daily basis to do things like create pop-up windows, drive dynamic HTML menus and validate forms. It cannot affect a user’s files or a server’s code. How much danger could there be in allowing someone to inject an unwanted script into a page?

An attacker could write nuisance scripts; a JavaScript with an infinite loop could render the victim’s browser unusable, forcing her to quit the browser. Similarly, the attacker could manipulate the window, by shrinking it, closing it or making it move randomly across the screen, or manipulate the Document Object Model to embed or alter text and images. A more sophisticated attack could use DOM manipulation to alter form values as part of an attempt to gather information intended for the vulnerable application; the form action could be switched to post the submitted data to a logging script on the attacker’s site, for instance. The original CERT advisory regarding XSS attacks described injection of HTML forms rather than scripts as a possible methodology. Form injection for password or credit card collection would be a natural extension of the hacker technique of using fake login pages to gather passwords. DOM manipulation via JavaScript would make this attack mechanism highly difficult to detect, but I am unaware of successful XSS attack using this methodology in the wild. However, if such an attack were well designed, it could be quite effective. For example, a carefully coded JavaScript could change the target of an e-commerce site’s legitimate checkout form without causing visible differences with the untampered site. If the new target were a page at attackersdomain.com that mimics the e-commerce site’s error page but logs credit card info, the injected script might go undetected, giving the attackers access to hundreds of credit card numbers.

An XSS attack could also use browser-specific vulnerabilities in scripting implementations to scrape information out of files on a user’s hard drive. Attackers might specifically target individuals with sensitive information stored on their local systems, sending poisoned URLs in e-mail designed to appeal to the specific intended victims.

The most common behavior of XSS attacks, however, is to gather cookies. Cookies are a technology initially designed for Netscape Navigator 1.0 to mitigate some of the problems stemming from HTML’s nature as a stateless protocol. They are small text files that reside on a user’s computer and store name-value pairs along with some metadata. Cookies are commonly used to store information intended to be persistent during a browser session or from session to session, such as session IDs, user preferences, or login information. The cookie specifications attempt to ensure that only the domain that set a cookie is allowed to access it.

Here is where the cross-site nature of XSS attacks comes into play. Because browsers are unable to distinguish between legitimate and injected scripts, they will treat the request as legitimate under JavaScript’s “Same-Origin Policy” and hand out the cookie information through standard JavaScript functions to any script on the vulnerable page. Since the text inside a <script> tag is not generally displayed, the victim may not even be aware that a script has executed. The injected script now has access to the user’s cookies and can pass them off to the attacker in any number of ways.

**QUESTIONS**

1. What are the key security risks in the LIB e-commerce environment?
2. Wherever possible, map the facts set out in the case to the detailed control objectives in the COBIT process DS5 ensure systems security. Identify evident control weaknesses.
3. Put yourself in the role of an external consultant on IT control systems and audit. The audit committee of LIB has become aware of LIB’s potential liability to XSS. The committee hired you on 1 August to provide a preliminary analysis on the vulnerability of LIB to XSS and on the current state of the security plan and security preparedness of LIB. Write a one-page memo (dated 10 August) to the audit committee. Make recommendations on:
   – The immediate handling of the XSS threat to LIB
   – Improvement in the management of security at LIB

Draw on the DS5 ensure systems security management and implementation guidelines in devising your recommendations. In writing your memo, consider the types of audit and inquiry procedures you might undertake in your investigation.

**SUGGESTED SOLUTIONS**

1. The key security risks of the e-commerce environment are:
   – Possible loss of customer data by malicious cross-scripting
   – Possible loss or corruption of data in transactions with other parties
   – Possible loss or corruption of data in caches of data
   – Possible external hacking attempts on poorly maintained e-commerce systems
   – Unknown effects of poorly designed or maintained scripting and ColdFusion code
2. In this section, facts in the case are mapped to relevant sections of DS5 (control objectives and control practices) and additional comments.
<table>
<thead>
<tr>
<th>Case Facts</th>
<th>Relevant Section of DS5</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Responsibility for security at LIB is shared by the operations manager,</td>
<td>DS5.1 manage security</td>
<td>Security planning is poorly organised and spread over three departments. There is co-ordination through the security management committee, but it is not well organised. There are major weaknesses in the design and implementation of the security planning. There is no overarching security plan.</td>
</tr>
<tr>
<td>program development manager and the network operations manager. Each group</td>
<td>measures</td>
<td></td>
</tr>
<tr>
<td>maintains a formal security plan. The three managers, together with the</td>
<td></td>
<td></td>
</tr>
<tr>
<td>director of IT and the director of internal audit, form a security task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>force that meets twice a year as the security management committee to</td>
<td></td>
<td></td>
</tr>
<tr>
<td>co-ordinate security developments and plans. The network operations group</td>
<td></td>
<td></td>
</tr>
<tr>
<td>includes a manager of network security, who is responsible for the</td>
<td></td>
<td></td>
</tr>
<tr>
<td>security of internal and external networks. Neither the operations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>manager nor the program development manager considers his/her group</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sufficiently large to warrant a full-time security person. The</td>
<td></td>
<td></td>
</tr>
<tr>
<td>responsibility for security is 40 percent of the job responsibilities of</td>
<td></td>
<td></td>
</tr>
<tr>
<td>the assistant manager of operations in the operations group and 30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>percent of the job responsibilities of the manager of quality in the</td>
<td></td>
<td></td>
</tr>
<tr>
<td>program development group. While no formal security plan is prepared,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>each of the three groups is required to report to the biannual meeting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>of the security management committee.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The JavaScript code used by LIB for its client interaction is derived</td>
<td>DS5.19 malicious</td>
<td>The management of JavaScript code from the local developer is clearly weak. Software and manuals are integrated—there is no mention of code maintenance or of stress testing the applications.</td>
</tr>
<tr>
<td>from a stock suite of e-commerce code sold by a local Australian e-commerce software developer. The e-commerce code is self-documenting. This means that all the help files and guidance for the integrator are incorporated into the body of the code. The software developer finds this more efficient than providing a series of manuals. Most clients do not use all the available code in the suite and may make alterations to the code to respond to the particular development environment. Indeed, the development team in the LIB program development IT group has made a series of code changes to the JavaScript to increase performance and meet particular business needs. These changes have also been self-documented.</td>
<td>software prevention, detection and correction</td>
<td></td>
</tr>
<tr>
<td>Case Facts</td>
<td>Relevant Section of DS5</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>Responsibility for security at LIB is shared by the operations manager, program development manager and the network operations manager. Each group maintains a formal security plan. The three managers, together with the director of IT and the director of internal audit, form a security taskforce that meets twice a year as the security management committee to co-ordinate security developments and plans. The network operations group includes a manager of network Security, who is responsible for the security of internal and external networks. Neither the operations manager nor the program development manager considers that their groups are sufficiently large to warrant a full-time security person. The responsibility for security in the operations group is 40 percent of the job responsibilities of the assistant manager of operations and 30 percent of the job responsibilities of the manager of quality in the program development group. While no formal security plan is prepared, each of the three groups is required to report to the biannual meeting of the Security Management Committee.</td>
<td>DS5.1: manage security measures</td>
<td>The planning of the security function at LIB is weak. It is spread across three groups and is managed only in an ad hoc fashion.</td>
</tr>
<tr>
<td>The report from each group (to the security management committee) should include a listing of security breaches and identified security vulnerabilities. Not every report from the group includes such detail and sometimes the reports are somewhat sketchy.</td>
<td>DS5.10 violation and security activity reports DS5.11 incident handling</td>
<td>The handling of incidents and reporting on the pattern of events are not timely (the meetings of the security management committee are only biannual) and the method of reporting is inconsistent. There seems to be no system for appropriate elevation of incidents.</td>
</tr>
<tr>
<td>The system uses a complex set of JavaScript codes to keep the transaction alive as the client clicks from page to page and interacts with LIB’s own back-end systems as well as the systems of the DMV, IIIC, etc. Some data that are transmitted to and from these third parties are cached in LIB’s databases and in temporary files in the LIB file systems, and some are passed directly through to the third party, without caching.</td>
<td>DS5.8 data classification</td>
<td>The handling of data in the LIB environment seems to be particularly weak. There is no discussion on explicit policies for caching of data, and the cache is handled in seemingly an inconsistent fashion.</td>
</tr>
<tr>
<td>Case Facts</td>
<td>Relevant Section of DS5</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>The security manager in the network group raises the possibility that LIB may be susceptible to XSS in her report to the security management committee. The manager also notes that it does not seem to be a vulnerability that affects the network operations group, as it does not impact on the availability or vulnerability of LIB’s Internet connectivity, intranet or other proprietary networks. The committee notes the report.</td>
<td>DS5.19 malicious software prevention, detection and correction</td>
<td></td>
</tr>
</tbody>
</table>

3. The student solution should be in the form of a memo to the audit committee. This gives the student practice in writing formal memos. The memo should cover some or all of the following points:
   - Scope of report
   - Nature of investigations
   - Potential vulnerability to XSS
   - Security management at LIB
   - Recommendations

The sample memo on the following page assumes that the investigation included a sample review of actual code; student assumptions on scope will vary.
MEMO

To: Chairperson, Audit Committee, LIB
From: [Insert name of student]
Date: 10 August
Re: Exposure of LIB to Cross-site Scripting (XSS) Attacks

This memorandum recommends committee action.

SCOPE
This memorandum is in response to the request of the audit committee, dated 1 August. As requested, we have investigated the vulnerability of LIB to cross-site scripting attacks and the overall system security environment at LIB. This memorandum does not constitute a report on a full audit of the internal controls in place at LIB. We have not undertaken a complete audit of the various systems, controls, development environment or of the application code at LIB.

In conducting this investigation, we have done the following:
1. Interviewed the following LIB staff including:
   • Network operations group
     – Network operations manager
     – Manager of network security
   • Operations group
     – Operations manager
   • Program development group
     – Program development manager
     – Quality manager
   • Internal audit
     – Director of internal audit
2. Undertaken program development and code review, including:
   • Sample reviews of program development plans for integration of ColdFusion and back-end systems
   • Sample reviews of JavaScript, PHP and ColdFusion code

EXPOSURE TO CROSS-SCRIPTING SECURITY ATTACKS AT LIB
We conclude that LIB is clearly at risk from XSS. The results of our study of the JavaScript, PHP and ColdFusion code in the production environment at LIB demonstrate poor quality assurance in the security management. There are a number of points in the information flow within LIB and amongst LIB and its partners, where an XSS hacker might be able to mount a successful XSS-based hijacking. The result of the XSS could result in the loss of critical customer information or hijacking of personal or corporate information.

THE SECURITY ENVIRONMENT AT LIB
We observe that there is an inappropriate management structure for the development and maintenance of security controls within LIB. Each of the functional units within the LIB IT function shares responsibility for the development and monitoring of controls. Co-ordination is weak, and the interunit committee meets only biannually. Further, there is no individual within the IT function with overall responsibility for security. Given the centrality of e-commerce systems to LIB and the complexity of the systems employed, functional staffing of security controls seems inadequate.

RECOMMENDATIONS
We make the following recommendations to the audit committee:
• Require the director of information technology to develop a new organisation model that removes overlapping responsibilities for security management at LIB and submit that model to the audit committee for review.
• Require the director of information technology to prepare a report for the audit committee within 30 days that details the processes undertaken to protect LIB from cross-scripting attacks.
• Require the director of information technology to present a staffing plan for security management and quality assurance within the IT function.
CASELET: PHOENIX ELECTRIC PROVIDER

CASE DESCRIPTION

Phoenix Electric Provider (PEP) is a large public power utility in the US. PEP distributes electrical energy to customers located north of Phoenix, Arizona. PEP was founded in 1939 as a publicly owned utility; PEP is wholly owned by its ratepayers and customers. PEP has revenues of more than US $600 million, and serves nearly 270,000 residential, commercial and industrial customers in a 700-square-mile service area. PEP does not generate power. It purchases electrical energy from the Western Area Power Administration, one of the large generators of electricity in the US, and distributes it to customers in the Phoenix area.

A five-member electric power board governs the PEP. The members of the board are appointed by the mayors of Phoenix, Mesa, Tempe and Scottsdale. The four city councils must confirm the board’s membership. Although a nonprofit business, PEP contributes to the economic health of local governments by making a large tax equivalency payment each year to the cities in the Phoenix metropolitan area.

PEP operates under a set of performance guidelines. For example, PEP makes the following “commitment to service” for retail clients:

1. Private lights will be repaired within five calendar days from the date reported by the customer or the customer will receive a US $10 credit.
2. Customer-requested service turn-on at an existing premise will be accomplished within 24 hours of promised date or PEP will waive the US $25 turn-on charge.
3. All routine new residential services to single-family and duplex homes will be connected within 72 hours after the electrical release is received by PEP from the Metro Codes Department (guarantee is five days if new poles are required) or PEP will waive the US $25 or $17 new-service connection charge.
4. Private light repair and service turn-on will be accomplished within the promised time frame, except during storm restoration.
5. All customer payments will be correctly posted to the customer’s account or the customer will receive a US $10 credit and any additional charges, if applicable, will be waived or paid by PEP.
6. Any disconnect made in error to a customer’s service (exclusive of storm and system outages) will result in immediate service restoration and a credit of US $25.
7. PEP will repair or reimburse the customer for damages in which PEP is at fault.

PEP uses many information systems and services to meet its overall strategic and operational objectives. Three of the most important elements of its IT infrastructure are:

1. The PEP information systems infrastructure consists of mainframe and client-server technology. Three primary mainframe applications are hosted at a service bureau. These applications are customer records, billing and credit management, and the geographic information systems (GIS) that map the transmission lines, underground trunking, substations, etc. The service bureau is responsible for the complete development and management of these three systems. The service bureau provides phone and computerised help desk services for PEP operations, customer support and engineering staff. The service bureau does not interface directly with PEP’s customers. There are multiyear outsourcing contracts between PEP and the service bureau. PEP’s own IT department operates all client-server systems. These systems support a wide range of accounting, customer support and real-time operations and specialist engineering services. The PEP client-server systems interface with the service bureau using IBM’s Websphere MQ messaging protocols.
2. Local UNIX and Windows platforms run several mission-critical systems. PEP operates the IBM RACF security system and employs various technology platforms, including UNIX, Novell NetWare and Oracle DBMS. These mission-critical systems are fundamental to service delivery. For example, reporting of customer outages and dispatch of outage restoration crews run on the computer-aided distribution operations system (CADOPS). These systems support the PEP customer service help desk that interfaces with customers. PEP’s customer service representatives handle more than 1.3 million calls a year in an average speed of less than 25 seconds per call. That means that each customer service representative talks to about 18,000 customers a year. In addition, about 75,000 customers come into the PEP offices every year to discuss everything from starting new service to billing issues. To ensure rapid answering of customer calls, PEP has installed an automated voice response unit, similar to those used by utilities and businesses across the country.

3. PEP is also a major user of the PeopleSoft human resource management systems to support payroll, human resource planning, benefits and training needs. PEP uses an integrated enterprise system from a leading software vendor to process all vendor transactions. One module processes all procurement activities. The financials module maintains all accounting and human resource information. There is a transfer of billing information from the customer system operated by the service bureau. The power grid operating systems provide data about customer consumption and electricity purchases. This is a separate suite of applications that are maintained by engineers in the operations department. The IT staff maintains all enterprise system applications. The enterprise system vendor provides upgrades several times a year on an as-needed basis. The software vendor system provides and updates documentation for the enterprise suite. The IT staff maintains the enterprise application integration software used to link the enterprise system with the power grid operating system. The staff’s work includes upgrades and associated documentation.

TEACHING NOTES

EXTRA INFORMATION

The PEP case is based on Nashville Electric Services (NES), which provides power utility services in the greater Nashville (Tennessee, USA) area. In search of new, more effective risk assessment, audit planning and review processes, NES began implementing COBIT in the late 1990s. Since then, NES has used COBIT to assess and manage risk and to develop all information systems audit programmes.

The audit department at NES first learned about COBIT some years ago through articles in technical publications. Not satisfied with the audit planning and review processes used at other organisations, the NES audit department supervisor wanted to implement a new, more effective audit methodology at NES. He was impressed with COBIT because of the control objectives and audit guidelines. Audit programmes should be based on an internationally accepted methodology. It is much easier to reach agreement on recommendations when they are based on audit guidelines that have worldwide acceptance.

Obtaining buy-in from senior management was an important achievement. The NES audit director has been very supportive of the COBIT methodology because it provides a consistent, credible benchmarking tool. COBIT enhances the ability to execute an audit plan that addresses the significant areas of risk. It also facilitates undertaking audit programmes that address the important security and control issues.

NES relies on COBIT primarily to develop audit programmes. The audit guidelines are used to develop the internal control questions and specific tests. Different technologies require different testing procedures; therefore, NES incorporates platform-specific guidelines into its testing methodology. The COBIT guidelines provided a consistent basis for developing those tests.
During the last three years, NES has used COBIT to develop all information systems audit programmes. These audit programmes have been developed, and NES plans on continuing to use the COBIT control objectives and audit guidelines to assess and manage risk and develop audit programmes that respond to the utility’s changing needs.

The case has been built around three of the COBIT processes. Each of the three technology and business scenarios set out in the case maps to a particular process. The outsourcing arrangements map to DS2 manage third-party services. The in-house mission-critical systems map to PO9 assess risks. The use of ERP systems maps to A12 acquire and maintain application software. All the necessary COBIT elements of these processes are included in the COBIT Student Book.

We suggest two alternative ways in which this case might be handled in the classroom. For alternative A, we provide a set of relatively structured questions that require the students to work with a particular process and its associated control objectives and audit guidelines. This approach builds student confidence in working with COBIT, but it does so in a directed fashion. Alternative B adopts a more open-ended approach by asking one group to design appropriate control systems to handle the risks set out in one of the technology and business scenarios set out in the case, while another group is asked to design audit work plans to test those controls. Later in the class time, the two groups come together and compare notes. There should not be significant differences in the outcomes from alternatives A and B—the differences come in the level of direction to the students.

We provide the alternative A and B case questions and some suggested solutions for question 1 of alternative A.

QUESTIONS

Alternative A Questions

You are the external auditor of PEP.

1. In relation to the outsourcing arrangements described in the case, employ the DS2 manage third-party services COBIT process to:
   a. Identify which of the DS2 control objectives together with their control processes are most appropriate for you to consider in designing your audit plan. Be prepared to justify your selection of the relevant control objectives.
   b. For three of the control objectives that you chose in (a), employ the COBIT audit guideline for DS2 to develop a draft set of audit procedures to provide assurance on meeting these control objectives.

2. In relation to the in-house mission-critical systems described in the case, employ the PO9 assess risks COBIT process to:
   a. Identify which of the PO9 control objectives together with their control processes are most appropriate for you to consider in designing your plan. Be prepared to justify your selection of the relevant control objectives.
   b. For three of the control objectives that you chose in (a), employ the COBIT audit guideline for PO9 to develop a draft set of audit procedures to provide assurance on meeting these control objectives.

3. In relation to the use of ERP systems described above, employ the A12 acquire and maintain application software COBIT process to:
   a. Identify which of the A12 control objectives together with their control processes are most appropriate for you to consider in designing your plan. Be prepared to justify your selection of the relevant control objectives.
   b. For three of the control objectives that you chose in (a), employ the COBIT audit guideline for A12 to develop a draft set of audit procedures to provide assurance on meeting these control objectives.
Alternative B Questions: Management
You or your group is to act as the chief information officer of PEP. You are to design a set of key controls needed to manage one of the technology and business scenarios set out in the case materials. The controls should include the skeleton of a staffing plan, including an organisation chart as well as at least eight fundamental controls. In developing these controls, you should rely on the most relevant COBIT control processes set out in the COBIT Student Book and the associated control objectives and management guidelines, all part of COBIT in Academia.

Alternative B Questions: Audit
You or your group is to act as the deputy director of internal audit and chief information systems auditor of PEP. You are to design an audit work programme to test the key controls over one of the scenarios set out in the case materials. In developing the audit programme, you should rely on the most relevant COBIT processes set out in the COBIT Student Book and the associated audit guidelines.

SUGGESTED SOLUTIONS
Alternative A Question 1
The appropriate COBIT process for the outsourcing technology and business scenario (1) is DS2 manage third-party services. The particular outsourcing relationship is a fundamental one that has been in operation for many years. The control objectives incorporated into DS2 that relate to the vetting or qualification of vendors do not seem to be relevant in this case.

The most relevant control objectives and practices of DS2 for the facts listed in (1) are:
- DS2.2 owner relationships
- DS2.5 outsourcing contracts
- DS2.6 continuity of services
- DS2.8 monitoring

A list of most significant control processes that PEP should have in place to manage the outsourcing arrangements with the service provider are:
- Appointment of a contract relationship manager with single point of responsibility for the relationship between PEP and the service bureau. Given the absolutely central role played by the outsourced IT functions (DS2.2), the relationship manager should be a relatively senior member of the PEP IT team.
- Development of procedures for formal management of the outsourcing agreement with the service bureau, including:
  - Formalisation of SLAs, including SLAs for problem management
  - Development of penalty structures that match to cost of catastrophic events, such as business interruption, or to lower-level failures, such as failure to provide adequate technical support to the in-house IT staff
  - Development of required performance levels and associated performance metrics to measure required levels
  - Development of processes to adjust service outputs to changes in PEP’s business operations (DS2.5)
- Development of a risk management and business interruption plan for business interruption caused by the actions of the service bureau (DS2.6)
- Development of procedures for monitoring service levels of the service bureau, including:
  - Application of performance metrics embedded in the SLA
  - Responsibilities for monitoring
  - Procedures for assessing variation against metrics
  - Procedures for layered escalation of problems diagnosed in monitoring service levels (DS2.8)