

Assignment 3: Mini-project; Information Systems Management System

A Security Case Study submitted in partial fulfillment of the requirements for the degree
of Doctor of Philosophy

Prepared By
Derek J. Sedlack
Division of Information Sciences
Graduate School of Computer and Information Sciences
Nova Southeastern University

Conducted for
DISS 799 Information Security
Dr. James Cannady, Professor



Certification of Authorship of Doctoral Work

Submitted to:

Professor James Cannady, Ph.D.

Student's Name:

Derek J. Sedlack

Date of Submission:

October 14, 2005

Purpose and Title of Submission:

Partial fulfillment requirement for doctoral degree.

Assignment 3: Mini-project

Certification of Authorship: I hereby certify that I am the author of this document and that any assistance I received in its preparation is fully acknowledged and disclosed in the document. I have also cited all sources from which I obtained data, ideas, or words that are copied directly or paraphrased in the document. Sources are properly credited according to accepted standards for professional publications. I also certify that this paper was prepared by me for this purpose.

Student's Signature: _____

A handwritten signature in black ink, appearing to read 'Derek J. Sedlack', is written over a horizontal line.

Java Application

Conduct a thorough analysis of your course topic through the development of a Java-based application that demonstrates some aspect of the topic area. Your task is to design and develop a prototype system that utilizes information security techniques to solve problems in an appropriate domain.

You are required to submit a final report on your application that includes the following:

1. Identify the problem that you were trying to solve.

Many security experts agree (Patterson and Blue, 2004; Chivers and Fletcher, 2005; McCumber, 1991; Peltier, 2000; Peltier, 2004) that the idea of top-down security aspects should be augmented by analyzing the entire organization, especially using a designed methodology like ISO 17799 in an Information Security Management System (ISMS). A number of companies have created manual lists of questions that cover the aspects of 17799, but they are lacking some essential elements that would allow a company to assess changes in policy or structure. Tying the automated analysis to collected comparisons should yield a baseline that either should be adopted, or would provide an aspiration for companies to reach at varying levels of adherence.

2. Justify the need for an information security-based approach to solve the problem.

Creating an applet that provides two separate measures in ISO 17799 implementations would provide companies with more flexibility and a more accurate assessment of their information security policies and measurements over time. The first aspect should be automated and stored. While the application does not database the information yet, future revisions would allow comparisons over time or inter-corporate comparisons for multi-national conglomerates. Dr. Cannady (2005) illustrated during lecture a graph of protection classes that provide an easily readable assessment of current measures for executive management, but how should the ISO 17799 assessment build such classes?

3. Discuss the problem solving methodology in detail.

During phase 1 of this applet, a number of questions from Praxiom (2005) were utilized in accordance with their non-commercial public agreement to create a preliminary list covering the ISO 17799 section for Organizational Policy Security. Digitizing the responses would allow easy future comparisons and allow a programmatic approach to build in comparison algorithms based on actuarial tables or corporate standards for assessment based on the responses. Tabulating responses from each section, or in total, will allow a raw or converted output that could lead to industry adopted comparisons or policy adherence that should enhance information security across the board. In phase 2, databasing

the information will allow longitudinal contrast/comparisons from within- or inter-corporate analysis. Cycles of sample surveys will also provide a baseline or fine tuning of the currently arbitrary algorithm.

4. Provide the design specifications for the system

The ISO 17799 documents provide the following security standard sections:

1. Security Policy (3)
2. Organizational Policy (4)
3. Asset Classification and Control (5)
4. Personnel Security Management (6)
5. Physical and Environmental Security (7)
6. Communications and Operations (8)
7. Information Access Management Control (9)
8. Systems Development and Maintenance (10)
9. Business Continuity Management (11)
10. Compliance Management (12)

Each Yes answer = 0 points

Each No answer = 1 point

Each N/A answer = 2 points

i = each answer in section 4 (Organizational Policy)

Q = total number of questions in section 4

S = the security result of the system analysis

$$S = \sum_i / Q$$

Valuations were conducted in the following manner:

Adequate protection: $S \geq 1.75$

Reasonable protection: $1.00 \geq S < 1.75$

Minimal protection: $.50 \geq S < 1.00$

Inadequate protection: $S < .50$

These values were arbitrarily assigned with no supporting evidence found during research, but based on the following logic: each N/A answer means the organization has not even considered this as a security risk; each No answer should be penalized; each Yes answer means that some form of security was implemented surrounding this question, or section. A high degree of N/A answers should countermand an equal number of Yes answers in that some number of areas were never identified for risk mitigation and could pose a catastrophic risk. A number of assessments should reveal a more accurate algorithm.

References

- Chivers, H., & Fletcher, M. (2005). Applying Security Design Analysis to a Service-Based System. *Software—Practice & Experience*, 35(9), 873-897.
- McCumber, J. (October 1991). Information Systems Security: A Comprehensive Model. *Proceedings 14th National Computer Security Conference*. National Institute of Standards and Technology, Baltimore, MD.
- Patterson, T. and Blue, S. G. (2004). *Mapping Security: The Corporate Security Sourcebook for Today's Global Economy*. Addison-Wesley Professional.
- Peltier, T. R. (2000). *Information Security Risk Analysis*. 1st. CRC Press, Inc.
- Peltier, T. R. (September 2004). Risk Analysis and Risk Management. *The EDP Audit, Control, and Security Newsletter*, 32(3), 1-17.
- Praxiom. (2005). *ISO IEC 17799 2000 Information Security Standard Translated into Plain English*. Retrieved from Praxiom research Group Limited on September 5, 2005 at: <http://praxiom.com/iso-17799-2000.htm>.

Appendix A: Java Source Code; Question.java

```
import javax.swing.*;
import javax.swing.event.*;
import javax.swing.text.*;
import javax.swing.BorderFactory.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import java.beans.PropertyChangeListener;
import java.beans.PropertyChangeEvent;
import java.text.*;
import java.lang.Integer;

public class Question
{

    private String questionText;
    private JRadioButton yes;
    private JRadioButton no;
    private JRadioButton na;

    public Question( String text )
    {
        questionText = new String ( text );
        yes = new JRadioButton ( "Yes" );
        no = new JRadioButton ( "No" );
        na = new JRadioButton ( "N/A", true );
    }

    public String getQuestionText()
    {
        return questionText;
    }

    public JRadioButton getYes()
    {
        return yes;
    }

    public JRadioButton getNo()
    {
        return no;
    }
}
```

```

public JRadioButton getNa()
{
    return na;
}

// return selected radio button
// -1 = unknown state
// 0 = yes
// 1 = no
// 2 = na
public int getSelected()
{
    if (yes.isSelected()) {
        return 0;
    }
    else if ( no.isSelected()) {
        return 1;
    }
    else if (na.isSelected()) {
        return 2;
    }
    else
        return -1;
}

}

```

Appendix A: Java Source Code (cont.); Security.java

```

/*****
*****/
/*                                     */
/* Created by Derek J. Sedlack          */
/* Address: 5043 Solar Point Drive      */
/*   Greenacres, Florida 33463         */
/* Email: Sedlack@Nova.edu             */
/* Web: scis.nova.edu/~sedlack         */
/*                                     */
/*****
*****/
/*                                     */
/* DISS 799 Information Security Management */
/* Dr. James Cannady                   */
/* Graduate School of Computer and Information Sciences */
/* Nova Southeastern University         */

```

```

/*                                     */
/*****
*****/
/*                                     */
/* This program should help provide a measurement of ISO IEC 17799 2000 (BS 7799)
*/
/* adherence through a list of questions designed to ensure some coverage of      */
/* area predefined by the ISO initiative. The section covered by this phase is    */
/* 04. Organizational Security                                                    */
/*                                     */
/* Future consideration should include all 10 sections and a more exhaustive      */
/* list of questions.                                                            */
/*                                     */
/*****
*****/
/*                                     */
/* This program is copyrighted and protected by U.S. Law and may not be used, copied,
*/
/* or distributed without the expressed permission of the author, partially or in total.*/
/*                                     */
/*****
*****/

```

```

import javax.swing.JOptionPane;
import javax.swing.JDialog;
import javax.swing.JButton;
import javax.swing.JRadioButton;
import javax.swing.ButtonGroup;
import javax.swing.JLabel;
import javax.swing.ImageIcon;
import javax.swing.BoxLayout;
import javax.swing.Box;
import javax.swing.border.Border;
import javax.swing.JTabbedPane;
import javax.swing.JPanel;
import javax.swing.JFrame;

import javax.swing.*;
import javax.swing.event.*;
import javax.swing.text.*;
import javax.swing.BorderFactory.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;

```

```
import java.text.*;
import java.lang.Integer;
```

```
public class Security extends JPanel implements ActionListener
{
```

```
    // Arrays used for Organizational Security Management
```

```
    private static String[] OrganizationalTitles = { "4. Organizational Security",
        "4.1 Establish a Security Infrastructure",
        "4.1.1 Set up an Information Security Forum",
        "4.1.2 Co-ordinate Security Implementations",
        "4.1.3 Allocate Security Responsibilities",
        "4.1.4 Set up Authorization Process for new Facilities",
        "4.1.5 Identify Specialized Security Advisors" };
```

```
    private static String[] OrganizationalList1 = {
        "Establish a management framework to control how your
organization implements information security.",
        "Establish a management forum to review and approve
your information security policy." ,
        "Establish a management forum to co-ordinate and
control the implementation of your organizations information security program.",
        "Establish a management forum to assign information
security roles and responsibilities.",
        "Ensure you have access to information security experts
and advisors within your own organization.",
        "Ensure your internal experts are able to provide
specialized information security advice.",
        "Ensure you have access to external information security
experts and advisors.",
        "Ensure your external advisors help you to monitor
changes in information security standards and methods.",
        "Ensure your external information security experts and
advisors help you to deal with security incidents.",
        "Ensure your organization encourages the use of a multi-
disciplinary approach to information security." };
```

```
    private static String[] OrganizationalList2 = {
        "Assign the responsibility for information security to a
single manager within your organization.",
        "Establish a management forum that you can use to
support information security initiatives.",
```

```
        "Ensure your security management forum promotes the
importance of information security.",
        "Ensure your security management forum ensures that
adequate resources are provided to support security.",
        "Ensure your security management forum reviews and
approves your information security policy.",
        "Ensure your security management forum reviews and
approves information security responsibilities.",
        "Ensure your security management forum monitors major
security threat changes and trends.",
        "Ensure your security management forum monitors how
exposed your information assets are.",
        "Ensure your security management forum monitors and
reviews information security incidents.",
        "Ensure your security management forum reviews and
approves improvements in information security." };
```

```
private static String[] OrganizationalList3 = {
        "Establish a management forum that you can use to co
ordinate the implementation of security controls.",
        "Ensure management forum members represent all
relevant areas of your organization.",
        "Ensure your security management forum distributes
information security roles and responsibilities throughout your organization.",
        "Ensure your security management forum reviews and
approves information security methods and techniques.",
        "Ensure your security management forum approves and
supports information security initiatives.",
        "Ensure your security management forum ensures that
security is considered during the information planning process.",
        "Ensure your security management forum evaluates the
adequacy of security controls used to protect new information systems or services.",
        "Ensure your security management forum reviews and
evaluates information security incidents.",
        "Ensure your management forum promotes the
importance of information security throughout your organization." };
```

```
private static String[] OrganizationalList4 = {
        "Define the responsibilities that control how individual
information assets should be protected.",
        "Define the responsibilities that control how information
security processes should be carried out.",
        "Define how specific information security roles and
responsibilities are distributed amongst various sites.",
        "Define how specific information security roles and
responsibilities are distributed amongst systems.",
```

"Define how specific information security roles and responsibilities are distributed amongst services.",
"Define how the responsibility for individual physical assets are allocated at the local level.",
"Define how the responsibility for individual information assets are allocated at the local level.",
"Define how the responsibility for individual security processes are allocated at the local level.",
"Ensure your information security manager has been given the responsibility for developing your security program.",
"Ensure your information security manager has been given the responsibility for implementing your security program.",
"Ensure your information security manager has been given the responsibility for identifying security controls.",
"Ensure asset owners have been given the responsibility for the security of their information assets.",
"Ensure your asset owners delegate specific security responsibilities to other managers or service providers.",
"Ensure asset owners ensure that delegated security responsibilities are clearly and completely stated.",
"Ensure delegated responsibilities for security assets and processes have been clearly and completely defined.",
"Ensure you document all delegated responsibilities for information security assets and processes.",
"Ensure you define and document all delegated authorization levels for security assets and processes.",
"Ensure your asset owners ensure that delegated security responsibilities are properly carried out." };

```
private static String[] OrganizationalList5 = {  
    "Establish a management authorization process to control new information processing facilities.",  
    "Ensure user managers approve of the purpose and authorize the use of all new information processing facilities.",  
    "Ensure your information security maintenance manager authorizes new information processing facilities.",  
    "Check new hardware to ensure that it will be compatible with existing system components.",  
    "Check new software to ensure that it will be compatible with existing system components.",  
    "Control the business use of personal information processing facilities.",  
    "Evaluate personal information processing facilities before they are used to process business information.",  
    "Authorize the use of personal processing facilities before they are used to process business information." };
```

```
private static String[] OrganizationalList6 = {
    "Identify an in-house information security advisor.",
    "Ensure your in house security advisor accumulates and
co-ordinates your organization's information security knowledge and experience.",
    "Ensure your in house information security advisor helps
your organization to make information security decisions.",
    "Ensure your in house information security advisor has
access to external security experts and advisors.",
    "Ensure your information security advisors have been
asked to provide advice on all aspects of information security.",
    "Have information security advisors been asked to assess
the security problems that threaten your organization.",
    "Ensure your information security advisors have been
asked to assess your organization's information security controls.",
    "Ensure information security advisors have direct access
to your organization's management personnel.",
    "Consult your security advisors whenever you have a
security incident or breach.",
    "Ask your information security advisors to investigate
security incidents or breaches." };
```

```
// Change the way the reporting chart worked
// a little with only one item being used right now
```

```
// The Framework consists of 10 segments
```

```
// 3. Security Policy
```

```
private String bSecPolX = "";
private String bSecPolXX = "";
private String bSecPolXXX = "";
private String bSecPolXXXX = "";
```

```
// 4. Security Organization
```

```
private String bSecOrgX = "";
private String bSecOrgXX = "";
private String bSecOrgXXX = "";
private String bSecOrgXXXX = "";
```

```
// 5. Asset Classification and Control
```

```
private String bAssetX = "";
private String bAssetXX = "";
private String bAssetXXX = "";
private String bAssetXXXX = "";
```

```
// 6. Personnel Security
```

```
private String bPerSecX = "";
```

```
private String bPerSecXX = "";
private String bPerSecXXX = "";
private String bPerSecXXXX = "";
```

```
// 7. Physical and Environmental Security
```

```
private String bPhySecX = "";
private String bPhySecXX = "";
private String bPhySecXXX = "";
private String bPhySecXXXX = "";
```

```
// 8. Communications and Operations Management
```

```
private String bCommMgtX = "";
private String bCommMgtXX = "";
private String bCommMgtXXX = "";
private String bCommMgtXXXX = "";
```

```
// 9. Access Control
```

```
private String bAccConX = "";
private String bAccConXX = "";
private String bAccConXXX = "";
private String bAccConXXXX = "";
```

```
// 10. System Development and Maintenance
```

```
private String bSysDevX = "";
private String bSysDevXX = "";
private String bSysDevXXX = "";
private String bSysDevXXXX = "";
```

```
// 11. Business Continuity Management
```

```
private String bBusMgtX = "";
private String bBusMgtXX = "";
private String bBusMgtXXX = "";
private String bBusMgtXXXX = "";
```

```
// 12. Compliance
```

```
private String bCompX = "";
private String bCompXX = "";
private String bCompXXX = "";
private String bCompXXXX = "";
```

```
// Wizard selection determinant
```

```
private JTabbedPane tabbedPane;
// Questions and option buttons
private Question[] q1 = null;
private Question[] q2 = null;
private Question[] q3 = null;
```

```

private Question[] q4 = null;
private Question[] q5 = null;
private Question[] q6 = null;

private JLabel info = null;
private boolean debug = true;

public Security()
{

    super(new BorderLayout());

    JComponent panel0 = makeInformationPanel();
    tabbedPane = new JTabbedPane(JTabbedPane.TOP,
JTabbedPane.SCROLL_TAB_LAYOUT);

    // Information Panel

    tabbedPane.addTab("Information", null, panel0, "");
    tabbedPane.setToolTipTextAt(0, "");

    //ALT + I to jump to Information Page
    tabbedPane.setMnemonicAt(0, KeyEvent.VK_I);

    JComponent panel1= makeOrganizationalPanel1();
    tabbedPane.addTab(OrganizationalTitles[1], panel1);
    //ALT + V to jump to Valuation Page
    tabbedPane.setMnemonicAt(1, KeyEvent.VK_E);

    JComponent panel2 = makeOrganizationalPanel2();
    tabbedPane.addTab(OrganizationalTitles[2], panel2);
    //ALT + A to jump to Assessment Page
    tabbedPane.setMnemonicAt(2, KeyEvent.VK_S);

    JComponent panel3 = makeOrganizationalPanel3();
    tabbedPane.addTab(OrganizationalTitles[3], panel3);
    //ALT + A to jump to Assessment Page
    tabbedPane.setMnemonicAt(3, KeyEvent.VK_C);

    JComponent panel4 = makeOrganizationalPanel4();
    tabbedPane.addTab(OrganizationalTitles[4], panel4);
    //ALT + A to jump to Assessment Page
    tabbedPane.setMnemonicAt(4, KeyEvent.VK_A);

```

```

JComponent panel5 = makeOrganizationalPanel5();
tabbedPane.addTab(OrganizationalTitles[5], panel5);
//ALT + A to jump to Assessment Page
tabbedPane.setMnemonicAt(5, KeyEvent.VK_U);

JComponent panel6 = makeOrganizationalPanel6();
tabbedPane.addTab(OrganizationalTitles[6], panel6);
//ALT + A to jump to Assessment Page
tabbedPane.setMnemonicAt(6, KeyEvent.VK_D);

JComponent panel7 = makeFrameWorkPanel();
tabbedPane.addTab("ISO 17799 FrameWork", panel7);
//ALT + V to jump to Valuation Page
tabbedPane.setMnemonicAt(7, KeyEvent.VK_F);

tabbedPane.setPreferredSize(new Dimension(1000, 600));

// Add the tabbed pane to this panel.
add(tabbedPane);

}

```

```

public void actionPerformed(ActionEvent e) {
    if ( tabbedPane.getSelectedIndex() == 0 ) {
        tabbedPane.setSelectedIndex(1);
    }
    else if ( tabbedPane.getSelectedIndex() == 1 ) {
        tabbedPane.setSelectedIndex(2);
    }
    else if ( tabbedPane.getSelectedIndex() == 2 ) {
        tabbedPane.setSelectedIndex(3);
    }
    else if ( tabbedPane.getSelectedIndex() == 3 ) {
        tabbedPane.setSelectedIndex(4);
    }
    else if ( tabbedPane.getSelectedIndex() == 4 ) {
        tabbedPane.setSelectedIndex(5);
    }
    else if ( tabbedPane.getSelectedIndex() == 5 ) {
        tabbedPane.setSelectedIndex(6);
    }
    else if ( tabbedPane.getSelectedIndex() == 6 ) {
        tabbedPane.setSelectedIndex(7);
    }
}

```



```

panel.add ( panel2 );
panel.add ( next, BorderLayout.PAGE_END );

return panel;
}

private JComponent makeOrganizationalPanel2()
{
    JPanel panel = new JPanel(new BorderLayout());
//    panel.setLayout(new BorderLayout());
//    create border around page with title
    panel.setBorder(BorderFactory.createTitledBorder(OrganizationalTitles[2]));

//    panel.setLayout(new FlowLayout(FlowLayout.LEFT,10,10));

    JPanel panel2 = new JPanel( new FlowLayout(FlowLayout.LEFT, 10, 10) );

    q2 = new Question[21];

    for (int x = 0; x < OrganizationalList2.length; x++) {
        q2[x] = new Question ( OrganizationalList2[x] );

        panel2.add ( new JLabel (q2[x].getQuestionText() ) );

        //Create the radio buttons.
        ButtonGroup group = new ButtonGroup();
        group.add( q2[x].getYes() );
        group.add( q2[x].getNo() );
        group.add( q2[x].getNa() );

        panel2.add (q2[x].getYes());
        panel2.add (q2[x].getNo());
        panel2.add (q2[x].getNa());

    }

    JButton next = new JButton ( "Next" );
    next.addActionListener(this);

    panel.add ( panel2 );
    panel.add ( next, BorderLayout.PAGE_END );
    return panel;
}

```

```

private JComponent makeOrganizationalPanel3()
{
    JPanel panel = new JPanel(new BorderLayout());
// panel.setLayout(new BorderLayout());
// create border around page with title
    panel.setBorder(BorderFactory.createTitledBorder(OrganizationalTitles[3]));

// panel.setLayout(new FlowLayout(FlowLayout.LEFT,10,10));

    JPanel panel2 = new JPanel( new FlowLayout(FlowLayout.LEFT, 10, 10) );

    q3 = new Question[21];

    for (int x = 0; x < OrganizationalList3.length; x++) {
        q3[x] = new Question ( OrganizationalList3[x] );

        panel2.add ( new JLabel (q3[x].getQuestionText() ) );

        //Create the radio buttons.
        ButtonGroup group = new ButtonGroup();
        group.add( q3[x].getYes() );
        group.add( q3[x].getNo() );
        group.add( q3[x].getNa() );

        panel2.add (q3[x].getYes());
        panel2.add (q3[x].getNo());
        panel2.add (q3[x].getNa());

    }

    JButton next = new JButton ( "Next" );
    next.addActionListener(this);

    panel.add ( panel2 );
    panel.add ( next, BorderLayout.PAGE_END );

    return panel;
}

private JComponent makeOrganizationalPanel4()
{
    JPanel panel = new JPanel(new BorderLayout());
// panel.setLayout(new BorderLayout());

```

```

// create border around page with title
panel.setBorder(BorderFactory.createTitledBorder(OrganizationalTitles[4]));

// panel.setLayout(new FlowLayout(FlowLayout.LEFT,5,5));

JPanel panel2 = new JPanel( new FlowLayout(FlowLayout.LEFT, 10, 10) );

q4 = new Question[21];

for (int x = 0; x < OrganizationalList4.length; x++) {
    q4[x] = new Question ( OrganizationalList4[x] );

    panel2.add ( new JLabel (q4[x].getQuestionText() ) );

    //Create the radio buttons.
    ButtonGroup group = new ButtonGroup();
    group.add( q4[x].getYes() );
    group.add( q4[x].getNo() );
    group.add( q4[x].getNa() );

    panel2.add (q4[x].getYes());
    panel2.add (q4[x].getNo());
    panel2.add (q4[x].getNa());

}

JButton next = new JButton ( "Next" );
next.addActionListener(this);

panel.add ( panel2 );
panel.add ( next, BorderLayout.PAGE_END );
return panel;
}

private JComponent makeOrganizationalPanel5()
{
    JPanel panel = new JPanel(new BorderLayout());
// panel.setLayout(new BorderLayout());
// create border around page with title
panel.setBorder(BorderFactory.createTitledBorder(OrganizationalTitles[5]));

// panel.setLayout(new FlowLayout(FlowLayout.LEFT,10,10));

JPanel panel2 = new JPanel( new FlowLayout(FlowLayout.LEFT, 10, 10) );

```

```

q5 = new Question[21];

for (int x = 0; x < OrganizationalList5.length; x++) {
    q5[x] = new Question ( OrganizationalList5[x] );

    panel2.add ( new JLabel (q5[x].getQuestionText()) );

    //Create the radio buttons.
    ButtonGroup group = new ButtonGroup();
    group.add( q5[x].getYes() );
    group.add( q5[x].getNo() );
    group.add( q5[x].getNa() );

    panel2.add (q5[x].getYes());
    panel2.add (q5[x].getNo());
    panel2.add (q5[x].getNa());

}

JButton next = new JButton ( "Next" );
next.addActionListener(this);

panel.add ( panel2 );
panel.add ( next, BorderLayout.PAGE_END );
return panel;
}

private JComponent makeOrganizationalPanel6()
{
    JPanel panel = new JPanel(new BorderLayout());
    // panel.setLayout(new BorderLayout());
    // create border around page with title
    panel.setBorder(BorderFactory.createTitledBorder(OrganizationalTitles[6]));

    // panel.setLayout(new FlowLayout(FlowLayout.LEFT,10,10));

    JPanel panel2 = new JPanel( new FlowLayout(FlowLayout.LEFT, 10, 10) );

    q6 = new Question[21];

    for (int x = 0; x < OrganizationalList6.length; x++) {
        q6[x] = new Question ( OrganizationalList6[x] );
    }
}

```

```

panel2.add ( new JLabel (q6[x].getQuestionText()) );

//Create the radio buttons.
ButtonGroup group = new ButtonGroup();
group.add( q6[x].getYes() );
group.add( q6[x].getNo() );
group.add( q6[x].getNa() );

panel2.add (q6[x].getYes());
panel2.add (q6[x].getNo());
panel2.add (q6[x].getNa());

}

JButton next = new JButton ( "Next" );
next.addActionListener(this);

panel.add ( panel2 );
panel.add ( next, BorderLayout.PAGE_END );
return panel;
}

private JComponent makeFrameWorkPanel()
{
    JPanel panel = new JPanel(false);

    panel.setLayout(new BorderLayout());
    panel.setBorder(BorderFactory.createTitledBorder("ISO 17799 Framework"));

    JPanel panel2 = new JPanel( new FlowLayout(FlowLayout.LEFT, 10, 10) );

    // JLabel info = new JLabel();
    info = new JLabel();
    String infoText = getTableData();

    info.setText(infoText);
    panel.add(info, BorderLayout.NORTH);

    JButton calculate = new JButton ( "Calculate" );
    calculate.addActionListener(this);

```

```

panel.add ( panel2 );
panel.add ( calculate, BorderLayout.PAGE_END );
return panel;
}

```

```

private String getTableData()
{
    String infoText = "<html>" +
        "<head>" +
        "<Body>" +
        "<Table align=center border=1 padding=0>" +
        "<TD></TD>" +
        "<TD colspan=4><Center>Protection Class</Center></TD>" +
        "</TR>" +
        "<TR>" +
        "<TD><CENTER>ISO 17799 Framework </center></TD>" +
        "<TD><CENTER> Class 1: <br>" +
        "  Inadequate <br>" +
        "  Protection </CENTER></TD>" +
        "" +
        "<TD><CENTER> Class 2: <br>" +
        "  Minimal <br>" +
        "  Protection </CENTER></TD>" +
        "" +
        "<TD><CENTER> Class 3: <br>" +
        "  Reasonable <br>" +
        "  Protection </CENTER></TD>" +
        "" +
        "<TD><CENTER> Class 4: <br>" +
        "  Adequate <br>" +
        "  Protection </CENTER></TD>" +
        "</TR>" +
        "<TR>" +
        "<TD> 3. Security Policy </TD>" +
        "<TD> <center></center> </TD>" +
        "<TD> <center>" + bSecPolX + "</center> </TD>" +
        "<TD> <center>" + bSecPolXX + "</center> </TD>" +
        "<TD> <center>" + bSecPolXXX + "</center> </TD>" +
        "</TR>" +
        "<TR>" +
        "<TD> 4. Organizational Security </TD>" +
        "<TD> <center>" + bSecOrgX + "</center> </TD>" +
        "<TD> <center>" + bSecOrgXX + "</center> </TD>" +
        "<TD> <center>" + bSecOrgXXX + "</center> </TD>" +
        "<TD> <center>" + bSecOrgXXXX + "</center> </TD>" +
        "</TR>" +

```

"<TR>" +
"<TD> 5. Asset Classification and Control </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bAssetX + "</center> </TD>" +
"<TD> <center>" + bAssetXX + "</center> </TD>" +
"<TD> <center>" + bAssetXXX + "</center> </TD>" +
"</TR>" +
"<TR>" +
"<TD> 6. Personnel Security </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bPerSecX + "</center> </TD>" +
"<TD> <center>" + bPerSecXX + "</center> </TD>" +
"<TD> <center>" + bPerSecXXX + "</center> </TD>" +
"</TR>" +
"<TR>" +
"<TD> 7. Physical and Environmental Security </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bPhySecX + "</center> </TD>" +
"<TD> <center>" + bPhySecXX + "</center> </TD>" +
"<TD> <center>" + bPhySecXXX + "</center> </TD>" +
"</TR>" +
"<TR>" +
"<TD> 8. Communications and Operations Management </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bCommMgtX + "</center> </TD>" +
"<TD> <center>" + bCommMgtXX + "</center> </TD>" +
"<TD> <center>" + bCommMgtXXX + "</center> </TD>" +
"</TR>" +
"<TR>" +
"<TD> 9. Access Control </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bAccConX + "</center> </TD>" +
"<TD> <center>" + bAccConXX + "</center> </TD>" +
"<TD> <center>" + bAccConXXX + "</center> </TD>" +
"</TR>" +
"<TR>" +
"<TD> 10. System Development and Maintenance </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bSysDevX + "</center> </TD>" +
"<TD> <center>" + bSysDevXX + "</center> </TD>" +
"<TD> <center>" + bSysDevXXX + "</center> </TD>" +
"</TR>" +
"<TR>" +
"<TD> 11. Business Continuity Management </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bBusMgtX + "</center> </TD>" +

```

"<TD> <center>" + bBusMgtXX + "</center> </TD>" +
"<TD> <center>" + bBusMgtXXX + "</center> </TD>" +
"</TR>" +
"<TR>" +
"<TD> 12. Compliance </TD>" +
"<TD> <center></center> </TD>" +
"<TD> <center>" + bCompX + "</center> </TD>" +
"<TD> <center>" + bCompXX + "</center> </TD>" +
"<TD> <center>" + bCompXXX + "</center> </TD>" +
"</TR>" +
"</Table>" +
"<Table width = 805 align=center border=0>" +
"<TR>" +
/* this does not make sense yet...
* will have to change appropriately when the
* measurements are solidified depending
* on reported coverage and specific items
* that require additional requirements
    "<TD>Legend</TD>" +
    "<TR>" +
    "<TD></TD>" +
    "<TD>no requirement</TD>" +
    "</TR>" +
    "<TR>" +
    "<TD>X</TD>" +
    "<TD>additional requirement based on implemented processes and
procedures</TD>" +
    "</TR>" +
    "<TR>" +
    "<TD>XX</TD>" +
    "<TD>additional requirement based on implementation of certified
products in at least half of the applicable product categories</TD>" +
    "</TR>" +
    "<TR>" +
    "<TD>XXX</TD>" +
    "<TD>additional requirement based on implementation of certified
products in all of the applicable product categories</TD>" +
    "</TR>" +
*/
    "</Table>" +
    "<body>" +
    "</html>" +
    "</head>";
return infoText;
}

```

```
/* Ok, this is truly hideous, but for this purpose it will work
 * If I don't change this for phase 2 I should be draw and shot
 * in public square in front of children in broad daylight ...
 */
```

```
private void calculateRisk()
{
```

```
float totalValues = 0, totalListItems = 0;
```

```
totalValues += calculate(q1);
totalValues += calculate(q2);
totalValues += calculate(q3);
totalValues += calculate(q4);
totalValues += calculate(q5);
totalValues += calculate(q6);
```

```
totalListItems += OrganizationalList1.length;
totalListItems += OrganizationalList2.length;
totalListItems += OrganizationalList3.length;
totalListItems += OrganizationalList4.length;
totalListItems += OrganizationalList5.length;
totalListItems += OrganizationalList6.length;
```

```
// See below for the twisted logic here
totalValues = totalListItems / totalValues;
```

```
if ( totalValues < .5 ) {
    bSecOrgX = "X";
    bSecOrgXX = "";
    bSecOrgXXX = "";
    bSecOrgXXXX = "";
} else if ( totalValues >= .5 && totalValues < 1.00 ) {
    bSecOrgX = "";
    bSecOrgXX = "XX";
    bSecOrgXXX = "";
    bSecOrgXXXX = "";
} else if ( totalValues >= 1.00 && totalValues < 1.75 ) {
    bSecOrgX = "";
    bSecOrgXX = "";
    bSecOrgXXX = "XXX";
    bSecOrgXXXX = "";
} else if ( totalValues >= 1.75 ) {
    bSecOrgX = "";
    bSecOrgXX = "";
```

```

    bSecOrgXXX = "";
    bSecOrgXXXX = "XXXXX";
}
info.setText(getTableData());

}
/**
 * Do calculations from all of the tabs (- the first and last)
 * to build our little chart
 * -1 means they didn't select anything (this is bad and they must be punished)
 * 0 means they have implemented the item
 * 1 means they did not implemented (at lease they are thinking about it, good for them!)
 * 2 means they are not even planning (again, punishment)
 *
 * Add up the values at this point for Dr. Cannady and determine
 * some meaningless arbitrary value that will later have real meaning
 * Well, make some algorithm anyway
 * n = number of questions for section
 * Ev = summation of the values for section
 * ex1. n = 10, 1/2 implemented = 0, 1/2 not = 5: 10/5 = 2
 * ex2. n = 10, 1/2 implemented = 0, 1/4 not = 2, 1/4 N/A = 6: 10/8 = 1.25
 * ex3. n = 10, 1/4 implemented = 0, 1/2 N/A = 10, 1/2 not = 3: 10/13 = 0.77
 *
 * I guess that lower numbers are a bad thing when determining using this method...
 *
 */
private int calculate(Question[] q)
{

    int totalValues = 0;

    if (q != null) {
        for (int i = 0; i < q.length; i++) {
            if (q[i] != null) {
                totalValues += q[i].getSelected();
            }
        }
    }

    return totalValues;
}

/**

```

```

* Create the GUI and show it. For thread safety, this method should be
* invoked from the event-dispatching thread.
*/
private static void createAndShowGUI()
{
    // Make sure we have nice window decorations.
    JFrame.setDefaultLookAndFeelDecorated(true);

    // Create and set up the window.
    JFrame frame = new JFrame("17799 Framework");
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    // Create and set up the content pane.
    JComponent newContentPane = new Security();
    newContentPane.setOpaque(true); // content panes must be opaque

    frame.getContentPane().add(newContentPane, BorderLayout.CENTER);

    // Display the window.
    frame.pack();
    frame.setVisible(true);
}
public static void main(String[] args)
{
    // Schedule a job for the event-dispatching thread:
    // creating and showing this application's GUI.
    javax.swing.SwingUtilities.invokeLater(new Runnable()
    {
        public void run()
        {
            createAndShowGUI();
        }
    });
}

}

```

Appendix B: Praxiom Legal Restriction Notice

Legal Restrictions on the Use of this Page

Thank you for visiting this page. You are, of course, welcome to view our material as often as you wish, free of charge. And as long as you keep intact all copyright notices, you are also welcome to print or make one copy of this page for your own personal, noncommercial, home use. But, you are not legally authorized to print or produce additional copies, or to copy and paste any of our material onto another web site. If you would like to purchase our material, please contact our Sales Desk. Our staff would be very pleased to take your order or to answer any questions you might have.