Integrative Programming and Technologies (IPT) – 23 core hours

IPT. Intersystems Communications
IPT. Data Mapping and Exchange
IPT. Integrative Coding
IPT. Scripting Techniques
IPT. Software Security Practices
IPT. Miscellaneous Issues
IPT. Overview of Programming Languages

Organizations typically use many disparate technologies that need to communicate and work with each other. A key component to the discipline of Information Technology is the integration of applications and systems. This knowledge area examines the various types of programming languages and their appropriate use. It also addresses the use of scripting languages, architectures, application programming interfaces and programming practices to facilitate the management, integration and security of the systems that support an organization.

IPT. Intersystem Communications

Minimum core coverage time: 5 hours

Topics:

Architectures for integrating systems
DCOM, CORBA, RMI
Web services and middleware
Network programming
Message and queuing services
Low level data communications

Core learning outcomes:

1. Describe and contrast the different types of architectures for integrating systems.
2. Define the role of DCOM, CORBA, and RMI in distributed processing.
3. Describe how web services are used to integrate disparate applications in an organization: for example, describe the role of the WSDL, SOAP, and UDDI architectures in creating and using web services.
4. Describe the role of socket programming in communicating between systems and contrast the protocols and uses of TCP/IP sockets and Datagram sockets.
5. Describe the purpose of message and queuing services and how they work and list the protocol used by one messaging service (e.g. JMS).
6. List commonly used low level data communications protocols (e.g., RS232), state conditions for when each protocol should be used, and outline the protocol for one low level communications protocol.

Advanced learning outcomes:

1. Create valid WSDL, SOAP and UDDI XML documents to define a web service. Write, debug, and test a web service. Deploy the web service to middleware and invoke the web service from an application across the network.
2. Design, develop and test a socket program that communicates between two different services using both TCP/IP sockets and Datagram sockets.
3. Design, develop and test a program that uses a messaging service to send asynchronous messages to another application across the network.

4. Design, develop and test a program that uses the RS232 protocol to communicate with a hardware device.

IPT. Data Mapping and Exchange

Minimum core coverage time: 4 hours

Topics:

Metadata
Data representation and encoding
XML, DTD, XML schemas
Parsing XML documents
XSL, XSLT and XPath

Core learning outcomes:

1. Define the term metadata.
2. Describe the characteristics of each of the following data encoding schemes, and recommend under what conditions each should be used: ASCII, EBCDIC, and Unicode.
3. Tell how XML and the document object model are being used to integrate and exchange data between systems.
4. Use DTD to create a document definition for a data structure: for example, given a DTD for data structure, create a XML document with real data.
5. Describe how XSL, XSLT and XPath are used to transform data streams.

Advanced learning outcomes:

1. Design, develop and test a program that converts a data stream using one encoding scheme to a different encoding scheme. (EBCDIC and ASCII)
2. Design, develop and test a program that uses SAX or DOM to parse an XML document.
3. Design, develop and test a program that uses XSL and XSLT to transform a data stream from one format to another.

IPT. Integrative Coding

Minimum core coverage time: 4 hours

Topics:

Design patterns
Interfaces
Inheritance

Core learning outcomes:

1. Define the importance of using design patterns and list the motivation for using each of the following design patterns: MVC, singleton, factory method, façade, proxy, decorator, and observer.
2. Describe what a programming interface is and why it is important to programming and give an example of where the use of a programming interface simplified the development of a system.
3. Define the concept of inheritance and describe how it can be applied to encourage code reuse.
4. Design an abstract class and use inheritance to create a class that extends the abstract class.
5. Design, develop and test an application that uses the abstract class.

Advanced learning outcomes:

1. Draw the UML class diagrams for each of the following design patterns: singleton, factory method, façade, proxy, decorator, and observer.
2. Design, develop and test a program that appropriately applies two or more design patterns to solve a problem.
3. Design and develop a programming interface. Implement the interface for at least two different realizations using the factory method pattern. Design, develop and test an application that uses the factory method pattern to instantiate objects for each realization and uses the common interface to access the functionality of each instance.

IPT. Scripting Techniques

Minimum core coverage time: 4 hours

Topics:

Scripting and the role of scripting languages
Creating and executing scripts
Influence of scripting on programming

Core learning outcomes:

1. Identify key scripting languages used for web scripting, server-side scripting and operating system scripting.
2. Write, debug and test a script that includes selection, repetition and parameter passing.

Advanced learning outcomes:

1. Write, debug and test a web page that uses scripting to validate the input values in a form.
2. Write, debug and test an interactive web based application that uses server-side script to process input from a web page.
3. Write, debug and test a script using an operating scripting language to facilitate the management of an operating system.

IPT. Software Security Practices

Minimum core coverage time: 4 hours

Topics:

Evidence-based security vs. code access security
Best security coding practices
Authentication to system resources and services
Encryption of data between systems and services

Core learning outcomes:

1. Contrast evidence-based security and code access security.
2. Define the goals of secure coding.
3. Give guidelines for authenticating and defining permissions to systems services and resources.
4. For each of the following “best secure coding” practices, give an example of a problem that can occur when the practice is not followed and then describe how to overcome the problem:
   a. Preventing buffer overflow
   b. Securing state data
   c. Securing method access
   d. Wrapper code
   e. Unmanaged code

f. Validation of user input
g. Remote considerations
h. Protected objects
i. Serialization
j. Robust error handling

Advanced learning outcomes:

1. Develop and test an application that uses systems services to encrypt a data stream, send it to a different system and decrypt the data stream.
2. Perform a security audit of the code of an existing system, identify problems that violate best security coding practices and make recommendations to fix each problem.

IPT. Overview of Programming Languages

Minimum core coverage time: 1 hour

Topics:

History of programming languages
Programming paradigms
Effects of scale
Virtual machines
Compiled vs. interpretative languages
Application vs. scripting languages

Core learning outcomes:

1. Contrast the differences between the structured and object-oriented programming paradigms.
2. Diagram and label models for both a compiled program and an interpretative program.
3. Describe the benefits and weaknesses associated with using a virtual machine.
4. Give an example where an application language and a scripting language would be more appropriate and give a valid reason to support your selection.

IPT. Miscellaneous Issues

Minimum core coverage time: 1 hour

Topics:

Adopt & adapt vs. make
Versioning and version control

Core learning outcomes:

1. List issues that should be considered when deciding whether to create new software or adapt existing software to solve a problem.
2. Tell why it is important to version software and describe one mechanism that can be used to control the versioning of software.

Advanced learning outcomes:

1. Use a version control system to create a new version of a software application, check out a module, make modifications to the module, check the module back in, and then rebuild, deploy and test the new version.
2. Install and set up a version control system for a new project.

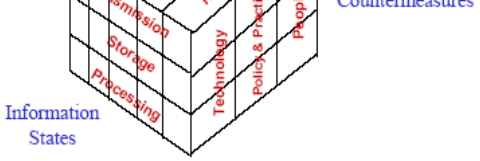
Information Assurance and Security (IAS) – 23 core hours

IAS. Fundamental Aspects
IAS. Security Mechanisms (Countermeasures)
IAS. Operational Issues
IAS. Policy
IAS. Attacks
IAS. Security Domains
IAS. Forensics
IAS. Information States
IAS. Security Services
IAS. Threat Analysis Model
IAS. Vulnerabilities

Since IT systems are increasingly under attack, knowledge of Information Assurance and Security (IAS) is of paramount importance to the profession of IT. The IT professional must understand, apply, and manage information assurance and security in computing, communication, and organizational systems. It is also important for the IT professional to provide users with a framework to be sufficiently security aware to be an asset to the organization rather than a liability. IAS includes operational issues, policies and procedures, attacks and defense mechanisms, risk analyses, recovery, and information security.

It should also be noted that many of the essential educational activities in this knowledge area may be illegal if performed outside a controlled environment, or without proper authorization. It is the responsibility of each individual program to appropriately administer these activities.

There is a model that we have found to be greatly helpful in communicating an IAS model in a relatively small amount of time and in a basic way. This model is shown in the diagram below.



Maconachy, Schou, Ragsdale, Welch: A Model for Information Assurance: An Integrated Approach (bibliography)

IAS. Fundamental Aspects

Minimum core coverage time: 3 hours

Topics:

History and terminology
Security mindset (reasoned paranoia)
Design principles (defense in depth)
System/security life-cycle

Security implementation mechanisms
    Gates, guards, guns; cryptography
Information assurance analysis model
    MSR model*; threats; vulnerabilities; attacks; countermeasures
Disaster recovery (natural and man-made)
Forensics

Core learning outcomes:

1. Briefly describe the history of the field of *Information Assurance and Security.*
2. Explain the relationship between threats, vulnerabilities, countermeasures, attacks, compromises and remediation.
3. Give examples of how IT system components (e.g. servers, routers, people, software) can be countermeasures, vulnerabilities, and also threats.
4. Explain the security mindset and the role of "paranoia" in that mindset.
5. Explain and give examples of why information assurance and security must be "built in" to design and architecture from the beginning to be most effective.
6. Outline the system life-cycle and its relationship to security.
7. Describe the Security Services as defined by the MSR model.
8. Describe the Information States as defined by the MSR model.
9. Describe the Countermeasures as defined by the MSR model.
10. Given the MSR model, explain how the components interrelate to categorize threats, vulnerabilities and attacks.
11. Describe a disaster recovery scenario.
12. Define forensics.
13. Describe a situation where a forensic investigation would be necessary.

Advanced learning outcomes:

1. Describe the major developments in the field of IAS and analyze the reasons for those developments.
2. Choose a development that has had significant impact on the field of IAS and explain the reason(s) why it is significant.
3. Prepare a threat analysis.
4. Given a threat analysis, recommend the appropriate countermeasures.
5. Demonstrate the ability to preserve evidence necessary for a forensic investigation.

IAS. Security Mechanisms (Countermeasures)

Minimum core coverage time: 5 hours

Topics:

Cryptography
    Cryptosystems
    Keys: symmetric & asymmetric
    Performance (software/hardware)
    Implementation
Authentication
    "Who you are, what you have, what you know"
    Bio-authentication (use of biometrics)
Redundancy
Intrusion detection

Core learning outcomes:

1. Explain the three key factors involved in authentication and how they are used to verify identity and grant access to a system.
2. Explain the process and value of two-factor authentication.
3. Explain the characteristics of an effective password to end-users.
4. Describe and compare physical access control to logical access control.
5. Identify the key types of biometric information utilized in authentication from the perspectives of accuracy, intrusiveness and efficiency.
6. Explain the differences between symmetric and asymmetric cryptosystems, e.g., number of keys required, the types of algorithms used, etc.
7. Explain what is meant by integrity, confidentiality, and authentication.
8. Explain how cryptosystems offer integrity, confidentiality and authentication.
9. Explain digital signatures and certificates.
10. Explain how public key infrastructure (PKI) works.
11. Use a PKI-based application to demonstrate how public-key cryptography works.

Advanced learning outcomes:

1. Describe the single sign-on authentication process and problems related to using and implementing this technology.
2. Compare and contrast key access control and authentication mechanisms to select the one appropriate to specific business contexts (Kerberos, RAS, etc,).
3. Compare the advantages and disadvantages of centralized access controls to decentralized access controls.
4. Describe the DES and 3DES algorithms.
5. Describe the AES algorithm.
6. Explain the differences between block and stream cryptosystems.
7. Explain the differences in efficiency and performance between software based and hardware based cryptosystems.
8. Install and configure a PKI-based application.

IAS. Operational Issues

Minimum core coverage time: 3 hours

Topics:

Trends
Auditing
Cost / benefit analysis
Asset management
Standards
Enforcement
Legal issues
Disaster recovery (natural and man-made)

Core learning outcomes:

1. Describe legal and ethical considerations related to the handling and management of enterprise information assets.
2. Specify what constitutes admissible evidence in a legal proceeding and how to acquire and maintain this information.

3. Describe the importance of and key elements involved in incident tracking to develop an incident handling and reporting process.
4. Identify risks associated with disasters or disruptions and specify key mitigation strategies.
5. Identify the types of company assets to be protected by a security plan.
6. Specify the key aspects of physical site security.
7. Describe the elements contributing to the cost of an organization's security management and operations process and their relation to risks and losses associated with information assurance or security related issues and incidents.
8. Describe and evaluate employment policies and practices that are relevant to safeguarding an organization's information assets.
9. Describe the importance of utilizing standards and key standard processes currently utilized in information assurance and their areas of relevance (i.e. DES – Data Encryption Standard).
10. Describe the purpose and elements of the key types of security audits. Discuss how various security standards (i.e. ISO 177799) impact the direction of these audits.

Advanced learning outcomes:

1. Develop an incident handling and reporting process.
2. Recommend appropriate operational and managerial processes to mitigate security and information assurance issues based on a Business Impact Analysis (BIA) report.
3. Create a BIA report that itemizes costs associated with information assurance and security activities and compares these costs to potential risks and costs associated with incidents in these areas.
4. Explain how key information security and assurance standards are or should be utilized in specific industry contexts.
5. Discuss the role of CASPR (Commonly Accepted Security Practices and Recommendations) forms in defining and approving standard operational and management practices.
6. Evaluate how changes in technology and the constantly changing threats impact operational and managerial practices and policies.

IAS. Policy

Minimum core coverage time: 3 hours

Topics:

Creation of policies
Maintenance of policies
Prevention
Avoidance
Incident response (forensics)
Domain integration (physical, network, internet, etc.)

Core learning outcomes:

1. Describe the role of policy and procedure in the IAS Model.
2. Explain why policy and procedure are listed as countermeasures.
3. Explain how poorly defined and executed policies can be a vulnerability.
4. Explain how an organization might develop a policy to defend against password vulnerabilities.
5. Explain why a password policy might need to be modified due to changing circumstances.
6. Explain why security policies must consider all aspects of an organization in order to be effective.
7. Give an example of how vulnerability in one area of an organization might enable a compromise in another area. (Example: weak physical security allows sniffer access to the LAN which allows a password to be read from a POP3 packet. The password is used to gain access to a corporate server.

Login access to the server allows a root-kit to be applied and the bad guy has total access to the server).

8. Describe a situation in which an incident would require a full forensic approach including evidence gathering, full chain of custody auditing and expert analysis.
9. Explain how failure to follow good forensic procedures could make prosecution of an attacker impossible.

Advanced learning outcomes:

1. Create a set of policies that implement a specified organizational objective.
2. Justify why a given policy is necessary to meet a specific organizational objective.
3. Update a set of policies to reflect a change in organizational objectives.

IAS. Attacks

Minimum core coverage time: 2 hours

Topics:

Social engineering
Denial of service
Protocol attacks
Active attacks
Passive attacks
Buffer overflow attacks
Malware (viruses, Trojan horses, worms)

Core learning outcomes:

1. Explain and give examples of the social engineering techniques used to gain access to computing and network assets in an organization.
2. Explain how a Denial of Service attack works against an organization's network.
3. List some different protocol attacks to which TCP/IP is susceptible.
4. Explain how the different protocol attacks (e.g. TCP/IP) work against an organization's network.
5. Explain some techniques used during an active attack.
6. Explain some techniques used during a passive attack.
7. Explain how an active attack might use information from a passive attack to compromise a system.
8. Describe and explain how a Buffer Overflow Attack might be used to compromise a system.
9. Identify and distinguish between the different types of Malware (viruses, Trojan horses, worms).

Advanced learning outcomes:

NOTE: Educational activities related to these outcomes should only be carried out in controlled circumstances and with appropriate authorizations.

1. Plan a social engineering attack against an organization's network.
2. Perform a penetration test (active attacks/passive attacks) against an organization's network.
3. Demonstrate the tools and technologies used in passive and active attacks.
4. Demonstrate a Denial of Service attack.

IAS. Security Domains

Minimum core coverage time: 2 hours

Topics:

Security awareness

Possible Domains:

Human-Computer Interaction
Information Management
Integrative Programming
Networking
Program Fundamentals
Platform Technologies
System Administration
System Integration and Architecture
Social and Professional Issues
Web Systems
Physical plant

Core learning outcomes:

1. Give examples of and explain shared concerns across a specified set of security domains.
2. Give examples of and explain concerns that are specific to specified security domains.
3. Discuss the impact of IAS on society and on one's professional and personal practice.

Advanced learning outcomes:

1. Describe the aspects of Human-Computer Interactions which must be considered in designing and implementing secure systems.
2. Explain how Information Management is impacted by security concerns.
3. Explain how Integrative Programming can utilize security and information assurance concepts to ensure the development of secure systems and applications.
4. List and describe several areas of Networking where security must be considered (i.e. access control, authentication, etc.).
5. Discuss several best practices in programming that can prevent security flaws.
6. Explain how security concerns may impact Platform section and operation.
7. Explain key security concerns from a Systems Administration perspective.
8. Discuss policies and practices that may be applied to Systems Integration and Architectures to ensure secure system operation and information assurance.
9. Identify the aspects of Web Systems that must be considered to ensure Web based applications are secure.
10. List the elements of the physical plant that should be considered in preventing security breaches.

IAS. Forensics

Minimum core coverage time: 1 hour

Topics:

Legal systems
Digital forensics and its relationship to other forensic disciplines
Rules of evidence
Search and seizure
Digital evidence
Media analysis

Core learning outcomes:

1. List three types of legal systems used by countries in the world.
2. Describe how digital forensics fits with the other forensic disciplines.

3. Describe a method for capturing a disk drive as evidence and proving its integrity for legal evidence purposes.
4. Explain the difference between the rules for a corporation seizing its property from an employee and law enforcement's seizing of property from a citizen.

Advanced learning outcomes:

1. Explain the concept of "reasonable expectation of privacy", its relationship to corporate policy banners displayed on workstations, and the use of email as evidence in the prosecution of an employee by the corporation.
2. List the areas on a disk that could contain evidence that are not accessible through normal operating system access.
3. List three methods for hiding information on a typical file system.
4. Explain why "free space" often contains interesting data.
5. Capture a forensic image of a disk drive and guarantee the integrity of the image using MD5 or SHA signatures.
6. Perform a basic media analysis of a captured drive using a forensic toolkit.

IAS. Information States

Minimum core coverage time: 1 hour

Topics:

Transmission
Storage
Processing

Core learning outcomes:

1. Give an example of a file representing a document in each of the three states.
2. Justify the statement "An electronic document is frequently in more than one state at the same time."
3. Describe a situation where the same document is in all three states at the same time.
4. Give examples of the relationships between the security services and information states.
5. Give examples of different vulnerabilities that apply to specific states.
6. Explain how the concept of "information state" relates to the concept of a "lifecycle".

Advanced learning outcomes:

1. Analyze a proposed data storage system describing all of the information objects and their states during the entire object lifecycle.
2. Describe some vulnerabilities of the information objects in each of the states.
3. Describe in detail the security services required to implement a specified security policy given an analysis as described above.

IAS. Security Services

Minimum core coverage time: 1 hour

Topics:

Availability
Integrity
Confidentiality
Authentication (source reliability)
Non-repudiation

Core learning outcomes:

1. Describe the possible availability levels for a web service.
2. Explain how redundancy and geographic dispersion relate to availability.
3. Explain the role of integrity, confidentiality, availability, authentication, and non-repudiation as security services.
4. Explain how one-way cryptographic functions are used to implement integrity in document transfer.
5. Explain how cryptographic encryption algorithms are used to implement confidentiality in document transfer.
6. Explain how one-way functions and encryption are used to implement a typical authentication service.
7. Explain how one-way functions are used to implement a non-repudiation service.

Advanced learning outcomes:

1. Describe the sequence of events for a typical Challenge Handshake Authentication Protocol (CHAP) authentication scheme.
2. Describe the sequence of events needed to protect the integrity and ensure non-repudiation of an electronic contract distributed through email to the participants in the contract.
3. Describe what needs to be done in addition to provide confidentiality for the contract in 2 above.

IAS. Threat Analysis Model

Minimum core coverage time: 1 hour

Topics:

Risk assessment
Cost benefit

Core learning outcomes:

1. Identify the aspects of a business that may be impacted by a security breach or interruption of operation.
2. Quantify the financial losses associated with potential security breaches and interruption of operations.
3. Identify and describe the nine steps to assess risks associated with security specified by the National Institute of Standards and Technology (NIST).
4. Describe the costs associated with actions that can be taken to mitigate security risks.

Advanced learning outcomes:

1. Perform a comprehensive risk assessment for a specified organization.
2. Create an Information Risk Management (IRM) policy for an organization by analyzing its information assets to determine the appropriate financial, managerial and security aspects of this policy.
3. Identify the quantitative and qualitative measures that can be used to assess risk and evaluate the effectiveness of risk management policies and practices.
4. Specify the value of the benefits to be achieved as a result of risk mitigation efforts and relate this benefit to the cost associated with achieving these benefits.
5. Justify appropriate mitigation strategies by comparing the costs associated with a specific risk and the mitigation strategy.

IAS. Vulnerabilities

Minimum core coverage time: 1 hour

Topics:

- Perpetrators
- Inside attacks
- External attacks
- Black hat
- White hat
- Ignorance
- Carelessness
- Network
- Hardware (design, implementation, installation, etc.)
- Software (design, implementation, installation, etc.)
- Physical access

Core learning outcomes:

1. Define white hat, black hat, hacker and cracker.
2. Explain how culture, community, tools, and technologies contribute to compromising systems.
3. Describe the role of the user in information assurance and how they fit into an overall information assurance plan for an organization.
4. Explain to a non-security community of users what measures they must follow and why, in a situation where their jobs are not security-related.
5. Give an example of how inside and external attacks are similar and are different.
6. List and explain the typical threats and vulnerabilities for an organization's network.

Advanced learning outcomes:

1. Demonstrate how you would test a system for vulnerabilities.
2. Perform a vulnerability analysis of a system.
3. Develop user education modules to educate users on their role in information assurance.
4. Choose a common vulnerability and describe how an attacker uses this vulnerability to gain access to a system.
5. Contrast how various technologies are used by hackers and crackers.
6. Compare and contrast various attack methodologies and differentiate between internal and external attacks.
7. Demonstrate how software contributes to the vulnerabilities of an organization.
8. Explain how design, implementation, and installation of software contribute to the vulnerabilities of an organization.
9. Demonstrate how hardware contributes to the vulnerabilities of an organization.
10. Explain how design, implementation, and installation of hardware contribute to the vulnerabilities of an organization.