# SCS5607 ETHICAL HACKING AND DIGITAL FORENSICS

# UNIT-IV

## ARCHITECTURE STRATEGIES

## Architecture strategies for computer fraud prevention

Components of an InformationTechnology Infrastructure

- The components of an information technology infrastructure and its security risks and controls are unique to every enterprise and particularly noteworthy in terms of how an architectural framework is designed and deployed to identify and mitigate the risks associated with insider computer fraud (ICF).
- Many organizations have Internet facing Web-based applications that can be accessed remotely by the insider either within the confines of the organization or remotely. Conversely, there are many applications within organizations that are not network based or Web based and function as stand-alone applications, which can also be used to perpetrate computer fraud.
- Consequently, the risk exposure for insider abuse is significantly higher for organizations that have Web-based versus traditional applications that can be accessed only within the organization.
    - Firewall.
        - Packet Filter Firewall
        - Packet Inspection Firewall
        - Application gateway Firewall
        - Circuit Level gateway
        - Proxy Server
    - Router
    - Host
    - Server
    - PC Workstation
    - Intrusion Detection systems.

***Architectural Strategies to Prevent and Detect ICF:***

In general, the industry lacks any best practices for application and IT infrastructure architectural design, and this presents some unique challenges in terms of knowledge sharing on this elusive topic.

When developing a system architectural design for an enterprise, there are several key considerations:

*Scalability*: This is the ease by which additional system processing capacity, throughput, or transactions can be increased (or decreased) over time.

*Replication*: Add processing resources that replicate and share part of the workload.

*Clustering*: Physically centralize but logically distribute the processing load.

*Locality of Decision-Making Authority*: Distribute the ability to affect modification, while centralizing the decision for which modifications to incorporate.

*Adaptability*: This is the ease by which the existing architectural design or configuration of a system can be updated to respond to changing conditions, performance congestion, security attacks, and so forth.

*Mitigating Architectural Mismatches*: This may occur when system components cannot interconnect to exchange data.

*Architectural Security Gaps*: There may be architectural security gaps that allow for unauthorized access and update capabilities to system resources.

*Component Based*: Configuring architecture using separable components.

*Multitier*: Configuring architecture into tiers that separate user interface from network access gateways (i.e., Web servers, security firewalls), from data storage/retrieval repositories.

*Types of System Architectural Designs for Information Processing:*

The primary types of system architectures for information processing include Service Oriented Architecture (SOA), distributive (client–server), and centralized information systems processing more commonly associated with mainframe and midrange computers.

Management’s decision to choose one or both of the architectural designs is a business decision and should be primarily based on the mission statement and the business goals and objectives of the enterprise.

### ***Service Oriented Architecture (SOA)***

The primary focus of this research is a SOA, which is essentially a collection of services. These services communicate with each other. The communication can involve either simple data exchange or it could involve two or more services coordinating some activity.

Web

services is the most likely connection of SOAs and uses eXtensible Markup Language (XML) to create a robust connection. A Web service is a software system designed to support interoperable machine-to-machine interaction over a network.

It has an interface described in a machine-processable format (specifically Web Services Description Language [WSDL]). Other systems interact with the Web service in a manner prescribed by its description using Simple Object Access Protocol (SOAP) messages, typically conveyed using Hypertext Transfer Protocol (HTTP).

An organization using Web services internally could easily have those services disrupted through the insider threat. Refer to the SOA diagram detailed in Figure 4.1

### *Centralized Processing*

This refers to the processing of all data at one single central location by a large mainframe computer. During the 1990s, the mainframe computer was in demand after a varied history. Mainframes became popular in the 1960s and 1970s because of their unprecedented computer power. During the 1980s and early 1990s, concepts such as client–server and distributed computing caused many to realize that although computing power could be purchased at a significantly lower capital cost, there were hidden costs involved.

### *Distributive Systems Architecture*

A Distributive Systems Architecture refers to any of a variety of computer systems that use more than one computer, or processor, to run an application. This includes parallel processing, in which a single computer uses more than one central processing unit (CPU) to execute programs. More often, however, distributed processin refers to local area networks (LANs) designed so that a single program can run simultaneously at various sites.

- Another form of distributed processing involves distributed databases—databases in which the data is stored across two or more computer systems.
- The database system keeps track of where the data is so that the distributed nature of the database is not apparent to users.

### *Client–Server Architecture*

This is a network architecture in which each computer or process on the network is either a client or a server. Servers are powerful computers or processes dedicated to managing disk drives (file servers), printers (print servers), or network traffic (network servers).

- Clients are PCs or workstations on which users run applications. Clients rely on servers for resources, such as files, devices, and even processing power.
- A newer client–server architecture, called a three-tier architecture, introduces a middle tier for the application logic. A special type of client–server architecture consists of three well-defined and separate processes, each running on a different platform:
- The user interface, which runs on the user's computer (the client).
- The functional modules that actually process data (this middle tier runs on a server and is often called the application server).
- A database management system (DBMS) that stores the data required by the middle tier (this tier runs on a second server called the database server).

The three-tier design has many advantages over traditional two-tier or single tier designs, the chief ones being as follows:

- The added modularity makes it easier to modify or replace one tier without affecting the other tiers.
- Separating the application functions from the database functions makes it easier to implement load balancing.

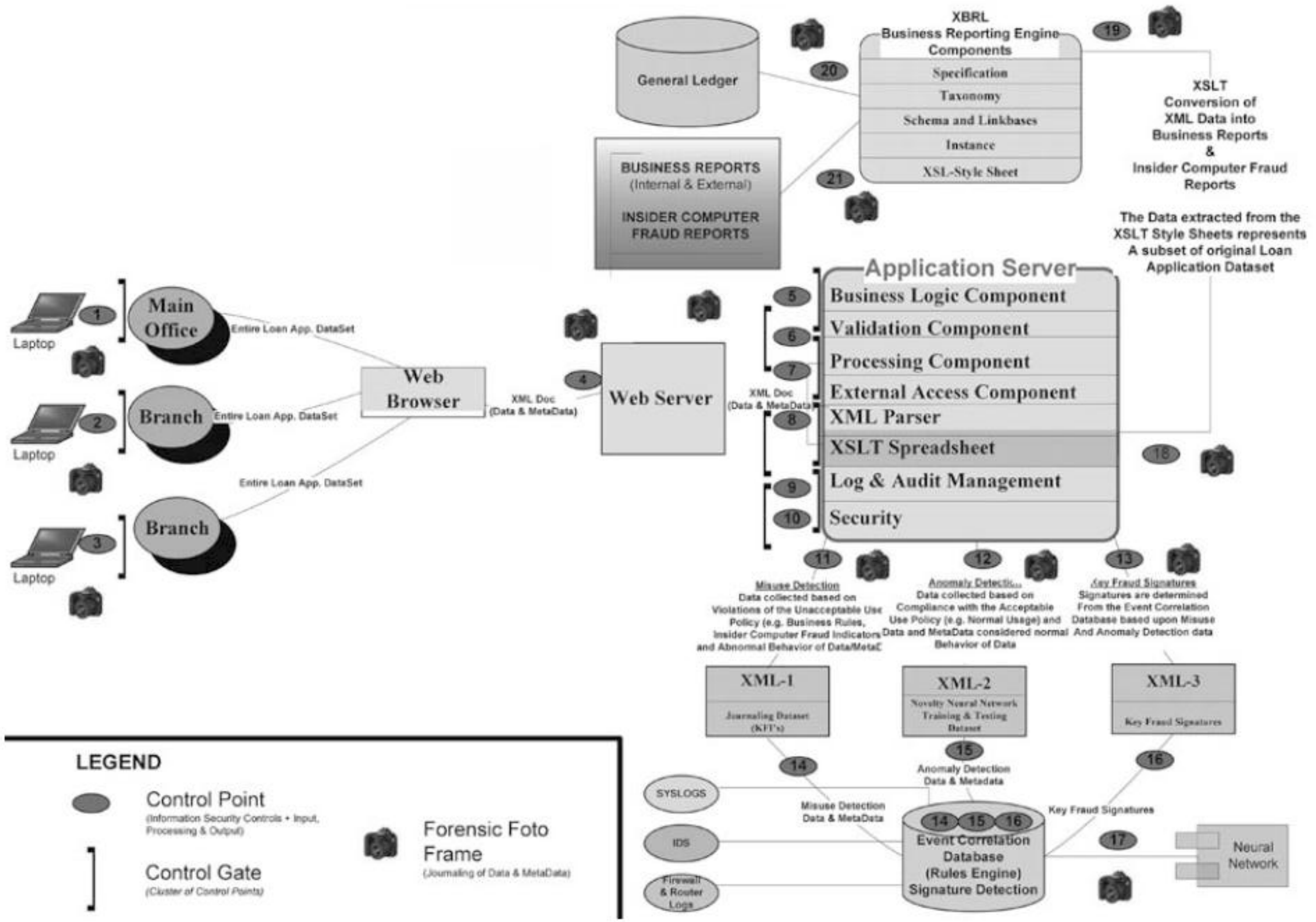

**Figure 4.1 Insider computer fraud Service Oriented Architecture (SOA).**

## Protection of Web Sites

The methods of detection of weak controls within an enterprise are introduced to allow the reader to gain a fundamental knowledge of what controls can be used within an enterprise to reduce ICF exposure.

Based on the results of the ICF taxonomy, there were several incidents involving former insiders who hacked into their former employers' systems. Although external hacking attacks by former employees occur less frequently than other forms of insider misuse, there were seven cases detected from research involving hacking activities, which points out the insider threat and the need to ensure that the appropriate level of safeguards and controls exist on network security.

## Intrusion Detection Systems

Intrusion detection systems (IDS) perform a number of different functions, including the following:

- Monitoring and analysis of user and system activity.
- Auditing of system configurations and vulnerabilities.
- Assessment of the integrity of critical system and data files.
- Recognition of activity patterns reflecting known attacks.
- Statistical analysis for abnormal activity patterns.
- Operating system audit trail management, with recognition of user activity
- reflecting policy violations.

## NIDS-Network Intrusion Detection Systems

Network intrusion detection systems (NIDS) use raw network packets as the primary data source. Additionally, the IDS uses a network adapter in promiscuous mode that listens and analyzes all traffic in real-time as it travels across the network.

The network IDS usually

has two logical components: the sensor and the management station. The sensor sits on a network segment, monitoring it for suspicious traffic.

The management station receives alarms from the sensors and displays them to an operator. The sensors are usually dedicated systems that exist only to monitor the network.

***Strengths***

- *Network Attack Detection*: The NIDS can detect some of the external and internal attacks that use the network, particularly new attack forms. A NIDS will alert for an external attack or an insider accessing the network remotely and wishing to conduct potential ICF activities.
- *Anomaly DetectorsDetection Ability*: Anomaly detectors have the ability to detect unusual behavior and therefore the ability to detect symptoms of attacks without specific knowledge of details.
- *Anomaly Detectors—Attack Signatures*: Anomaly detectors have the ability to produce information that serves as the basis for the development of new attack signatures.
- *No Modifications of Production Servers or Hosts*: A network-based IDS does not require modification of production servers or hosts.
- *No Production Impact*: NIDS will generally not negatively impact any production services or processes, because the device does not function as a router.
- *Self-Contained*: NIDS runs on a dedicated system that is simple to install, generally plug-and-play after some configuration changes, and then is set to monitor network traffic.

***Weaknesses***

- *Limited to a Network Segment*: NIDS examines only network traffic on the segment to which it is directly connected. It cannot detect an attack that travels through a different network segment.
- *Expensive*: The problem may require that an organization purchase many sensors
- in order to meet its network coverage goals.
- *Limited Detection*: Detection is limited to programmed attacks from external sources; however, it is not considered effective for detecting the more complex information threats.
- *Limited Coordination*: There is little coordination among sensors, which creates a significant amount of analysis traffic.
- *Difficulty with Encryption*: A network-based IDS may have a difficult time handling attacks within encrypted sessions.

- *False Positives*: Anomaly detection generates a significant number of false positives.

***Host-Based Intrusion Detection Systems(HIDS)***

Host-based IDSs operate on information collected from within a computer system, which gives this HIDS a distinct advantage over NIDS, due to the fact that it is easy for a target to see the intended outcome of the attempted attack compared to network attack.

The host-based IDS looks for signs of intrusion on the local host system. These frequently use the host operating system audit trails and system logs as sources of information for analysis.

Every platform is different in terms of what system audit trails and system log reports are produced; however, in Windows NT, there are system, event, and security logs, and in UNIX there are Syslog and other operating-specific log files.

This IDS architecture generally uses rule-based engines for analyzing activity; an example of such a rule might be, superuser privilege can only be attained through the su command.

***Strengths:***

- A host-based IDS usually provides much more detailed and relevant information than a network-based IDS.
- Host-based systems tend to have lower false-positive rates than do networkbased systems.
- Host-based systems can be used in environments where broad intrusion detection is not needed, or where the bandwidth is not available for sensorto-analysis communications.
- Finally, a host-based system may be less risky to configure with an active response, such as terminating a service or logging off an offending user.

***Weaknesses:***

- Host-based systems require installation on the particular device that you wish to protect.
- If the server is not configured to do adequate logging and monitoring, you have to change the configuration of, possibly, a production machine, which is a tremendous change management problem.
- Host-based systems are relatively expensive.

- They are almost totally ignorant of the network environment. Thus, the analysis time required to evaluate damage from a potential intrusion increases linearly with the number of protected hosts.

### ***The Penetration Testing Process:***

Penetration testing is an emerging practice used in organizations to attempt to identify and test their vulnerabilities before a malicious agent has the opportunity to exploit it. The various techniques presented here attempt to penetrate a target network from a particular frame of reference, both internal and external to the organization.

- The Pen testing process and objective at its most fundamental level is trying to emulate a hacker by assessing the security strengths and weaknesses of a target network.
- A comprehensive Pen test will surface and test for real-world vulnerabilities (for example, open services) detected during the network security scan.
- One of the primary goals in Pen testing is to identify security vulnerabilities in a network and to assess how an outsider or insider to an enterprise may either deny service or gain access to information to which the attacker is not authorized.

#### *Methodology*

Determining what type of Pen test to perform will be largely attributed to the threat management is trying to replicate and how the test should be conducted (internal versus external) and who should be conducting the test. Different types of Pen tests are as follows:

- *Internal (Consultant Scenario)*: Typically, during an internal Pen test, an attempt will be made to emulate a consultant with zero knowledge of the entity; however, the person making the attempt will possess a sufficient skill level to perform the ethical hack but will have no access rights to the network other than physical access.
- *Internal (Internal Employee Scenario)*: Another threat profile that should be tested within an internal threat scenario would be that of an internal employee.

An internal Pen test searches for potential security vulnerabilities within the internal (trusted) network.

- *External*: External Pen tests are intended to identify vulnerabilities that were established through the organization connection to the Internet via a firewall or gateway.
- Fundamentally, external Pen tests are designed to test the adequacy of the perimeter defense mechanisms, such as remote access controls, firewalls, encryption, intrusion detection, and incident response.
- *Pen Teams*: The Pen Test teams are performed by the Final Four accounting firms and other consulting firms. The term “Tiger team” is typically used to describe Pen testing teams. The term “Spider team” is commonly used to refer to those individuals involved in Pen testing against Web servers that are located outside of the enterprise’s network.

**Web Services-Reducing Transaction risks:**

*Web Services Security for a Service Oriented Architecture:*

- Web services were selected as the primary architecture based upon its growing use within the marketplace. However, the concept of the Forensic Foto Frame, which captures data at various control points and control gates along the path of a particular transaction, would apply in a distributed client–server environment equally as well.
- The primary players in the Web services space include Microsoft, IBM, Oracle,BEA, and others. There is concern in the marketplace that reliance on vendor alliances is questionable, which may lead to vendor politics, thereby preventing a single, consistent set of standards. The front-runners in Web services design and development tools vendors are Microsoft VS.net, BEA WebLogic Workshop, and IBM/Rational Websphere.

*Major Groups Involved in Establishing Standards for Web Services Security:*

- *OASIS (Organization for the Advancement of Structured Information Standards)*:A global consortium that drives the development and adaption of e-business standards. OASIS produces worldwide standards for security, Web services, XML conformance, business transactions, electronic publishing, topic maps, and interoperability within and between marketplaces. Key standards include extensible rights markup language, WS-Security,

Security Assertion Markup Language (SAML) provisioning, biometrics, and eXtensible Access Control Markup Language (www.oasis-open.org).

- *W3C (World Wide Web Consortium)*: Develops interoperable technologies (specifications, guidelines, software, and tools). Key standards include XML encryption, XML signature, and XKMS (XML Key Management Specifications) (www.w3c.org).
- *Liberty Alliance*: The Liberty Alliance project was formed in 2001 to establish an open standard for federated network identity. Key standards include SAML to pass standards-based security tokens between identity and authentication systems (www.projectliberty.org).
- *WS-I (Web Services Interoperability Organization)*: An open-industry organization chartered to promote Web services interoperability across platforms, operating systems, and programming languages. The organization works across all industries and standards organizations to respond to customer needs by providing guidance, best practices, and resources for developing solutions for Web services (www.ws-i.org).

### ***Web Services Security—Technical Security Concerns:***

The technical security governing Web Services continues to evolve and mature (i.e.,SAML). There are some organizations that are moving slowly into Web services, by deploying such activity internally as a pilot prior to engaging directly into external e-commerce and data transmissions.

- *Transmission of Executables*: Web services allow the hidden transmission of executables and other malicious content. Web services allow for the transmission of any kind of data within the flow of message transactions.
- *Cyber Attacks*: Web services security vendors and other perimeter security vendors have not yet achieved a significant level of sophistication with regar to the aforementioned types of attacks.

### *Security Assertion Markup Language (SAML)*

SAML is an XML vocabulary used to convey trustworthy, digitally signed authentication and user credential information between applications or domains and independent from actual authentication mechanisms, user directories, or security policies.

- *Browser-Based SSO*: The browser-based SSO usage of SAML allows an identity provider to transparently pass identity attributes to service providers as part of a SSO environment. The underlying mechanics of SAML allow the service provider to validate that the information has integrity.
- *SOAP-Based Web Services*: A SAML assertion can be directly attached to the header of a SOAP envelope to indicate who the end user is behind any particular transaction.
- *Other Web Services Security Proposals*: There are other efforts underway to evaluate the security aspects of Web services, which include evaluating a Web services proof of concept, which involves interoperability testing between Java and .NET platforms across financial firewalls. There are extensions of SOAP being designed to provide data confidentiality, integrity, authentication, and message reliability.

*Specific Types of Web Services Security Solutions:*

- *Security Infrastructure*: A number of existing standards can be used to enhance the security of Web services. Although some like SSL and HTTPS can be used on their own to provide point-to-point security, all of the following can be incorporated in the more elaborate schemes currently being developed.
- *SSL (Secure Sockets Layer)*: SSL is a network encryption and authentication mechanism developed by Netscape, which is used in HTTPS. Currently a de facto standard, SSL has been submitted to the Internet Engineering Task Force (IETF) for standardization, which will take the form of Transport Layer Security (TLS) Protocol.
- *TLS (Transport Layer Security)*: A Transport Layer Security Protocol was adopted by IETF as RFC 2246. TLS is based on SSL, which it may eventually supersede.
- *HTTPS*: HTTPS is a secure form of HTTP, implemented by the use of SSLinstead of plain text. Data is encrypted in both directions, and the server is authenticated by an SSL certificate
- *XML Digital Signature*: This is a set of XML syntax and processing rules for creating and representing digital signatures (a necessary building block for the WS-Security Framework). As well as its obvious purpose of guaranteeing

message integrity, XML Digital Signature can be used to implement nonrepudiation.

- *XKMS (XML Key Management Specification)*: A specification submitted toW3C by Microsoft, VeriSign, and webMethods in March 2001, XKMS aims to provide a Web service public key infrastructure (PKI) interface in such a way as to hide as much complexity as possible from the client..
- *Message-Level Authentication*: This includes HTTP basic and HTTP digest.
- *Security Assertion Markup Language (SAML)*
- *X509*
- *Data-Element Level Encryption and Digital Signatures*
- *Ability to Secure*: There is the ability to secure SOAP messages and plain XML messages.
- *XML Application Firewall*: An XML application firewall fits well as a noninvasive
- drop-in solution in front of any XML or Web services interface.

*Extensible Markup Language (XML)*

XML provides a context by which applications can send and receive messages and documents that describe the nature of their content in machine-readable form.

Web services will define how data will be requested and passed, with the standards incorporated in an interface within an existing application, which will allow any other application with a similar interface to connect with it and to exchange data. Web services are sometimes referred to as “XML in motion” because Web services protocols define the transport mechanisms for XML-based communications.

There are two formats for defining XML document declarations: XML document- type definition (DTD) and XML Schema. The XML DTD was the original representation that provided a road map to the syntax used within a given class of XML document. Because XML tags are not predefined as in HTML, a DTD describes tag names and distinguishes the required and optional elements that may appear in a document.

*XML and Security.*

At the current time, encrypting a complete XML document, testing its integrity, and confirming the authenticity of its sender is fairly straightforward; however, at times it may be necessary to use encryption and authenticate in arbitrary sequences and involve different users and originators. At the present time, the most important sets of developing specifications in the area of XML-related security are XML encryption, XML signatures, XACL (eXtensible Access Control Language), SAML, and XKMS.

XML encryption will allow encryption of digital content, such as GIF, SVG, or XML fragments, while leaving other parts of the XML document not encrypted.

*Simple Object Access Protocol (SOAP)*

SOAP is an XML-based protocol for document-based messaging and remote procedure calls across distributing computing environments. SOAP-based messages are transport independent: designed for use over HTTP, SOAP also can be used with other transport mechanisms, such as the Simple Mail Transfer Protocol (SMTP) that may be required when traversing corporate firewalls.

A SOAP message has three sections: a general message container, called an envelope, that is used to specify the encoding style of the message data; the body containing the actual message data; and an envelope header designed to carry additional transaction-specific information, such as authentication,
routing, and scope detail.

*SOAP and Security.*

- When securing SOAP messages, various types of threats should be considered:
- The message could be modified or read by antagonists or an antagonist could send messages to a service that, while well-formed, lacks appropriate security claims to warrant processing.
- Based on the OASIS Web Services Security Model, which applies the use of message security tokens combined with digital signatures to protect and authenticate SOAP messages. Security tokens assert claims and can be used to assert the binding between authentication secrets or keys and security identities.Protecting the message content (confidentiality) from being disclosed or modified without detection (integrity) is a primary security concern.

### *Problems with Web Services Security.*

There are no industry best or even sound processes or practices for risk mitigation for Web services threats and vulnerabilities within the financial services sector or other industries.

Web services are based on standards that are simple and portable but provide no built-in security mechanisms. Therefore, data transferred via Web services can be exposed to threats. Security standards for Web services are being developed that attempt to define a standard set of message headers to achieve integrity and security, but they are not yet available.

The three primary risk categories that apply to Web services security include authentication, authorization, administration, and cyber-security concerns.

The risk assessment process should include, at the minimum, the following general categories involved in Web services:

- *Authentication*: Authentication is a core requirement of a secure Web services architecture—only free and public Web services do not require some form of authentication. The major choices for authentication include how credentials are bound to the Web services request, the type of credentials, whether a session
- token is used, and where and how authentications are processed.
- *Authorization*: Web services architects must decide on the richness of authorization policy and where and how authorization is performed.
- *Richness of Authorization*: The business situation drives authorization policy requirements. This can range from simple implied entitlement to complex policy-based and instance-based authorization.

*Administration*

- Determining the appropriate Web services administration is a security challenge.A comprehensive secure Web services architecture involves securing more than an independent, stand-alone Web services tier, which in turn involves multiple software infrastructure elements, each with its own security.
- For example, a business partner security infrastructure integration (federation) provides emerging solutions and standards for federated security.
- A security infrastructure is needed to enable business partners (or divisions within a single enterprise) to recognize and trust each other's users.

## References:

1. Kenneth C.Brancik, "Insider Computer Fraud", Auerbach Publications Taylor & Francis, Group 2008.

2. Ankit Fadia, "Ethical Hacking", Second Edition Macmillan India Ltd, 2006