About the IPPF

The International Professional Practices Framework® (IPPF®) is the conceptual framework that organizes authoritative guidance promulgated by The IIA for internal audit professionals worldwide.

**Mandatory Guidance** is developed following an established due diligence process, which includes a period of public exposure for stakeholder input. The mandatory elements of the IPPF are:

- Core Principles for the Professional Practice of Internal Auditing.
- Definition of Internal Auditing.
- Code of Ethics.
- International Standards for the Professional Practice of Internal Auditing.

**Recommended Guidance** includes Implementation and Supplemental Guidance. Implementation Guidance is designed to help internal auditors understand how to apply and conform with the requirements of Mandatory Guidance.

**About Supplemental Guidance**

Supplemental Guidance provides additional information, advice, and best practices for providing internal audit services. It supports the Standards by addressing topical areas and sector-specific issues in more detail than Implementation Guidance and is endorsed by The IIA through formal review and approval processes.

**Practice Guides**

Practice Guides, a type of Supplemental Guidance, provide detailed approaches, step-by-step processes, and examples intended to support all internal auditors. Select Practice Guides focus on:

- Financial Services.
- Public Sector.
- Information Technology (GTAG®).

For an overview of authoritative guidance materials provided by The IIA, please visit [www.globaliia.org/standards-guidance](http://www.globaliia.org/standards-guidance).
About GTAGs

Within the IPPF’s Supplemental Guidance, Global Technology Audit Guides (GTAGs) provide auditors with the knowledge to perform assurance or advisory services regarding an organization’s information technology (IT) and information security (IS) risks and controls. The Standards that give rise to the GTAGs are listed below.

- **1210.A3** – Internal auditors must have sufficient knowledge of key information technology risks and controls and available technology-based audit techniques to perform their assigned work. However, not all internal auditors are expected to have the expertise of an internal auditor whose primary responsibility is information technology auditing.

- **2110.A2** – The internal audit activity must assess whether the information technology governance of the organization supports the organization’s strategies and objectives.

- **2130.A1** – The internal audit activity must evaluate the adequacy and effectiveness of controls in responding to risks within the organization’s governance, operations, and information systems regarding the:
  - Achievement of the organization’s strategic objectives.
  - Reliability and integrity of financial and operational information.
  - Effectiveness and efficiency of operations and programs.
  - Safeguarding of assets.
  - Compliance with laws, regulations, policies, procedures, and contracts.

- **2220.A1** – The scope of the engagement must include consideration of relevant systems, records, personnel, and physical properties, including those under the control of third parties.

About Professional Toolkits

The GTAGs will be supplemented by Professional Toolkits, which will dive deeper into the respective subject areas and provide practical guidance for understanding IT-IS risks and controls, as well as planning and executing audits. Each GTAG will have a Toolkit, and each Toolkit will consist of three types of guidance:

- **Basic** – Mainly consisting of common IT-IS process narratives, flowcharts and diagrams, to illustrate the business processes, risks, and controls.

- **Certified** – Further discussion of risks and controls shown in the Basic-level diagrams and narratives, to help with scoping and building an audit program. There will be plenty of references to specific controls and sub-controls in widely-used frameworks.

- **Executive** – The Audit Executive Center will produce high-level subject overviews, case studies, recorded videos, and other content to satisfy the needs of Chief Audit Executives.

Commented [CJ1]: Deborah asks: should this be toolkits or tools, based on what we have now (Risk Audit Tool for AEC, for example)?
Table of Contents

Executive Summary ...................................................................................................................................... 3
Introduction ............................................................................................................................................ 4
  Objectives ........................................................................................................................................ 5
Business Application Audit Planning ................................................................................................... 5
  Scoping the Audit ............................................................................................................................. 7
Performing the Audit ............................................................................................................................ 8
  Technology Planning ......................................................................................................................... 8
  Software Development Life Cycle .................................................................................................... 9
  Production Support .......................................................................................................................... 11
  Other Relevant Control Types ......................................................................................................... 13
Conclusion .......................................................................................................................................... 15
Appendix A. Relevant IIA Standards and Guidance ........................................................................... 16
Appendix B. Glossary .......................................................................................................................... 17
Appendix C. References ....................................................................................................................... 23
Acknowledgements .............................................................................................................................. 24
Executive Summary .............................................................................................................................. 3
Introduction ........................................................................................................................................ 4
  Objectives ....................................................................................................................................... 5
Business Application Audit Planning ................................................................................................... 5
  Scoping the Audit ............................................................................................................................. 7
Performing the Audit ............................................................................................................................ 8
  Technology Planning ......................................................................................................................... 8
  Software Development Life Cycle .................................................................................................... 9
  Production Support .......................................................................................................................... 11
  Other Relevant Control Types ......................................................................................................... 13
Conclusion .......................................................................................................................................... 15
Appendix A. Relevant IIA Standards and Guidance ........................................................................... 16
Appendix B. Glossary .......................................................................................................................... 17
Appendix C. References ....................................................................................................................... 23
Acknowledgements .............................................................................................................................. 22
Executive Summary

Business applications may be a single software program or a collection of hardware, firmware, and software applications operating as an integrated system to enable the organization’s processes. Business applications are subject to many common IT and information security (IS) control categories, each of which consists of standard control processes of varying relevance, depending on the organization’s and application’s specific circumstances.

Controls over business applications can be loosely categorized as relating to:

1. **Technology planning** – IT-IS planners work with business unit leaders to design technology solutions to meet business needs. Enterprise and security architects determine requirements for applications and component technologies, often documented in a technology roadmap. Planning for component obsolescence is a critical step in the roadmap.

2. **Software (or systems) development life cycle (SDLC)** – Applications require coding that adheres to functional and security requirements. The source code is written, tested, released into service, and revised as needed to fix errors, address security flaws, accommodate new technology, or add features.

3. **Production support** – In-service business applications are configured for use and supported by system administrators, who are usually in IT. System administrators work with the benefitting business units to create system roles for various job functions and implement account authorization, reauthorization, and deactivation processes.

4. **Application security** – Controls over secure design and coding, patch management, user access management, and event logging are part of planning, the system software development life cycle, and support processes.

Other significant control objectives over business applications include but are not limited to:

5. **Records and information management (RIM)** – Maintaining documentation of application architecture, system interfaces, data flows, and source code.

6. **Vendor management** – Ensuring applications purchased from or significantly modified by vendors are covered by contracts with sufficient assurance of performance and security.

7. **Asset management** – Maintaining an inventory of in-service applications and related metadata to support various governance and operational needs.

8. **Database administration** and **business intelligence** – Controlling access to and use of application data to support privacy and management reporting objectives.

Stakeholders such as senior management and the board’s audit committee require assurance services to verify whether controls over business applications are well-designed and effectively implemented.
Introduction

Some features that distinguish a business application include whether the software has been programmed to perform or enable specific business processes and whether user accounts have differentiated permissions. Common examples of business applications include enterprise resource planning systems, point-of-sale systems, industrial control systems, customer relationship management and billing systems, and many others. A tool, like a spreadsheet or database application, can be engineered to essentially become a business application, but for internal audit purposes, a basic spreadsheet or database application usually would not be considered a business application.

Typically, the administration of business applications is performed by the entity’s IT department; however, it is not uncommon for shadow IT functions to exist within other business units, especially as vendor-managed and cloud-based applications become more prevalent. Regardless of the department performing system administration and oversight, the benefitting business unit personnel have roles to play in defining business needs, executing authorization controls, and providing feedback on system performance.

Some commonly used IT-IS control frameworks include ISACA’s COBIT 2019, the National Institute of Standards and Technology’s (NIST’s) special publication 800-53 revision 5 (NIST SP 800-53r5), and the Center for Internet Security (CIS) Top 20 controls for cybersecurity. IT-IS personnel frequently benchmark operational and security controls against one or more of these frameworks, and although each framework uses its own groupings of controls, there are substantial commonalities among the frameworks in terminology and categorization. This GTAG references the guidance in these frameworks, where doing so may be helpful to an auditor. Readers of this guide are assumed to have a general knowledge of IT-IS risks and controls, as described in the GTAG IT Essentials for Internal Auditors, and are encouraged to incorporate a review of the full texts of one or more IT-IS control frameworks in their audit planning and test programs.

As directed by a risk-based audit plan, internal auditors will examine how organizations develop or acquire business applications to ensure adequate security, service delivery, and availability. Auditing a business application involves a risk assessment, a scoping analysis, and tests to evaluate control frameworks cited.

Control Frameworks Cited

Three external IT-IS control frameworks (there are many others) are mentioned in this Guide, and each provides much more information about specific controls than is discussed here. (See Appendix C. References for details on these sources.) Internal audit teams are encouraged to review common IT-IS control guidance to understand common risks and controls during audit planning and fieldwork.
the design and implementation of relevant controls and determine whether any significant risk exposures exist. It is beneficial for internal audit, IT-IS teams, and the benefiting business unit personnel to collaborate, as each can provide valuable insight into inherent risks, the strength of controls, and residual risks. Following this approach, an internal auditor will demonstrate adherence to Standard 1220 — Due Professional Care, which states: “Internal auditors must apply the care and skill expected of a reasonably prudent and competent internal auditor.”

Objectives

This Guide will help the reader:

- Define business applications and obtain a working knowledge of relevant processes, including related governance and security controls.
- Understand risks and opportunities associated with business applications.
- Understand components of the system software development life cycle, including:
  - Coding to meet operational and security requirements.
  - Testing for security and functionality.
  - Controlling the release and storage of source code.
  - Installing security patches and other updates.
- Understand components of production support, including:
  - Configuring the application and connections to external systems.
  - Administering system roles and accounts.
  - Responding to user and operational feedback, including errors and outages.
- Understand some of the related controls for documentation, vendor management, asset management, and reporting from application databases.
- Understand the basics of auditing business applications, including specific controls that need evaluating.

Business Application Audit Planning

To assess the inherent risks of a business application, it may be helpful to ask a series of questions about the context of its use and other factors about its administration. A common set of questions with relevance to an application risk assessment is shown below.
<table>
<thead>
<tr>
<th>Question</th>
<th>Relevance to Risk Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Is the application a single software program or a platform that consists of multiple applications and products interfacing and operating together?</td>
<td>1. An application platform is inherently riskier than a single application due to security and performance risks with component technologies, in addition to the primary application.</td>
</tr>
<tr>
<td>2. What significant business processes does the application support, and who is/are the business owner(s)?</td>
<td>2. Unclear or disengaged ownership may lead to suboptimal management of strategy, governance, business requirements, service delivery, or the prioritization or funding of enhancements.</td>
</tr>
<tr>
<td>3. Does the application process significant financial transactions, and is it a significant input to the financial statement accounts?</td>
<td>3. Applications that process financial transactions may be subject to external, financial statement audits, so coordination of assurance should be considered. Fraud is also a likely risk.</td>
</tr>
<tr>
<td>4. Does the application process or store personally identifiable information (PII), or other sensitive data?</td>
<td>4. Data privacy, compliance, and cybersecurity risks are heightened when an application handles or stores personally identifiable information.</td>
</tr>
<tr>
<td>5. Is the application internet-facing?</td>
<td>5. Controls over encryption, firewalls, secure coding, connections to external systems, and monitoring controls are typically more important for internet-facing applications.</td>
</tr>
<tr>
<td>6. To what extent are vendors involved in developing, administering, or hosting the application?</td>
<td>6. Controls related to the software development life cycle, system administration, database administration, and hosting may be carried out by non-employees, heightening vendor risk and the importance of performance oversight, as well as identity and access management.</td>
</tr>
<tr>
<td>7. To what extent is the application included in standardized control processes, like identity and access management, patch management, monitoring for availability, and cybersecurity, etc.?</td>
<td>7. Due to control inheritance, an application that is covered by standardized controls covering multiple systems may not need to have those controls tested again in an internal audit of the application.</td>
</tr>
<tr>
<td>8. Has the application been the subject of an internal audit before? If so, what were the findings, overall risk rating, and scope of the audit? Are there any open corrective actions?</td>
<td>8. Previously identified design or implementation findings can be retested, although a fresh risk assessment and scoping analysis is advisable.</td>
</tr>
<tr>
<td>9. Has the application been covered by a recent risk assessment performed by IT-IS, or an external audit, e.g., a Payment Card Industry Data Security Standards (PCI-DSS) or Service Organization Controls (SOC) audit? Were any deficiencies or excessive risk exposures noted?</td>
<td>9. Considerations are similar to #8, although coordination of assurance activities and a largely advisory approach to the engagement may reduce the client's audit fatigue.</td>
</tr>
<tr>
<td>10. Is the application (and its component applications) included in the organization's software inventory, with key metadata — like business owner; production support, development, and security contacts; system criticality and data classification ratings; and system interfaces, etc. — completely populated?</td>
<td>10. If the organization does not have an inventory system integrated with governance, development, and support processes, there is a higher likelihood of manual processes that are inherently riskier than automated ones.</td>
</tr>
</tbody>
</table>
Combining an understanding of the importance and delivery model of the application and its context — with an assessment of risk-relevant attributes — should give the internal auditor enough information to begin considering an engagement’s scope.

**Scoping the Audit**

Business applications may enable, support, or monitor business processes — which themselves may be part of larger processes. They also rely on a number of controls that may be standardized throughout the enterprise or tailored to the specific circumstances of the application under review. Therefore, the scope of a business application audit is determined by consideration of the context, risks, and engagement objectives.

Deciding which business processes or controls to include in the audit scope is typically driven by the engagement objective. An integrated audit of operational and technical controls may be desirable; however, only the audit’s treatment of the business application controls would be covered by this Guide.

**Business Process Scoping Method**

The business process scoping method evaluates all the systems that support a particular business process. The focus of such an audit would likely be on the functionality of the application or its ability to meet business needs. However, the scope could include other aspects — such as vendor management or identity and access management — based on an engagement risk assessment. The internal auditor identifies the input, processing, and output systems within the audit subject as part of the scoping activity. Sometimes, especially for complex business applications like an enterprise resource planning system, different modules of the same ecosystem can be similar to input or output applications. Therefore, it may be important to identify the application modules that support the business process and the data that flows between them.

**Single Application Scoping Method**

The single application scoping method is used to review the controls within a single business application, as opposed to taking a business process scoping approach. Such an audit could comprise an end-to-end view of the application ecosystem, including technology planning, software development life cycle, production support, application security, record and information management, vendor management, asset management, and database administration controls,
among others. This approach might be preferred for business applications — for example, industrial control systems — that support processes whose operational controls are likely to be covered in separate audits.

**Single Module Scoping Method**

Sometimes, an audit of functionality in a single module — such as the fixed assets module in an accounting application — may be desired. These audits are narrow in scope, primarily focusing on whether business rules are documented and properly implemented. As such, a single module audit would likely focus on application functionality controls and the working relationship between the benefitting business units, production support, and database administration teams.

**Performing the Audit**

The scoping decisions will determine which control types are relevant to the audit. Common risks and controls for each type are described in the following sections.

**Technology Planning**

High-level planning controls enable the service relationship between IT-IS and other business units and ensure that business applications are compatible with existing and future technologies used in the enterprise. In the COBIT 2019 framework, technology planning controls are primarily described in the Evaluate, Direct and Monitor and Align, Plan and Organize domains. The NIST SP 800-53r5 framework is oriented more towards security than service management; however, planning-related controls can be found in the Planning, Program Management, PII Processing and Transparency, Risk Assessment, System and Services Acquisition, and System and Communications Protection control families.

**Gathering Requirements and Build vs. Buy**

The business owner(s) and benefitting business units are responsible for identifying a need for a business application and for determining the capabilities needed to support business processes. An IT or IS leader can also be a business owner for applications that meet their department’s needs — for example, to support an IT service desk, or for event log monitoring and analysis processes.

IT leaders help the business owner determine whether the organization should develop the software internally, engage external developers, purchase commercial, off-the-shelf software, or seek vendor-provided solutions — which could include cloud-based, on-premises, or hybrid hosting models.

If an organization decides to develop software internally using employees or contractors, IT leaders typically engage with the benefitting business units to identify how the software needs to enable the business processes, with limiting parameters and automation where appropriate. If the organization purchases an off-the-shelf or vendor-provided solution, the benefitting business unit needs to verify that the software provides the necessary capabilities, or even decide to alter processes to match the software’s functionalities.
Security in Design

When planning the system design, security-related attributes to be considered include the system’s security categorization, whether and where to deploy encryption, the risks associated with potential vendors, cybersecurity risks, and more. If the application is going to be connected to the internet, the placement and configuration of web application firewalls needs to be determined. An audit of these processes typically verifies whether the system architecture and data flow documentation includes IS measures.

System Software Development Life Cycle

Application development is characterized as a life cycle because the process is usually circular: Software is planned, developed, tested, and implemented, and then operational feedback is obtained, which informs further planning and development, and so on. There are many languages and development methods used to create software programs; however, the sections below will focus on the generalized objectives and controls in creating applications that meet business needs.

Software Development

Controls over software development are described in COBIT 2019 objective BAI03 Managed Solutions Identification and Build; in the NIST SP 80053r5 System and Services Acquisition and System and Communications Protection control families; and in CIS Top 20 control 18 Application Software Security. The controls in each framework range from high level to detailed design of solution components, which can be developed or procured. Important considerations for development include establishing a separate coding environment, which should not have any direct interface to the production environment, and preventing developers from having access to production code or systems to prevent the insertion of unauthorized code or security bypass mechanisms — or to otherwise subvert the authorized operations of the application.

Static and dynamic code testing tools are often used in a test environment to improve the quality, efficiency, and security of software code. Restricting developers from testing their own code is part of maintaining a separation of duties, which is necessary to mitigate the risk of intentional or unintentional vulnerabilities.

Application Functionality Controls

Organizations manage operational risks through the use of programmed or configurable controls, such as enabling a three-way match control and acceptable variance tolerances for invoices going through the accounts payable process. The IIA and others have historically referred to these types of controls as “application controls,” but to be more specific, they can be thought of as application functionality controls because they enable business processes through data input validation, separation of business functions, balancing of processing totals, transaction logging, and error reporting. The functions are programmed according to the business owner’s documented
requirements, known as business rules. These controls are usually preventive or detective, but they may also enable forensic analysis.

Types of application functionality controls include:

- **Input controls** – Used mainly to check the integrity of data entered into a business application to ensure that it remains within specified parameters, is limited to valid data types, and is properly authorized.
- **Processing controls** – Ensure processing is complete, accurate, authorized, and timely.
- **Output controls** – Ensure accuracy and completeness by comparing output results to inputs and properly recording output data.
- **Integrity controls** – Monitor data in process and at rest to ensure it remains consistent and persistent.
- **Transaction and event logging** – Assign unique identifiers (IDs) to transactions and events to enable forensic investigation and ensure accountability.

In engagements involving application functionality controls, internal auditors evaluate whether these controls are documented and implemented appropriately, due to their importance to operations. One way to do that is to examine the design and results of user acceptance tests; another is to evaluate the root cause analysis process for performance issues, verifying whether frequent or high-impact events led to configuration or code changes.

**Benchmarking Application Controls in Audits of Controls over Financial Reporting**

Appendix B of the U.S. Public Company Accounting Oversight Board’s (PCAOB) Auditing Standard No. 5, “An Audit of Internal Control Over Financial Reporting That is Integrated with An Audit of Financial Statements,” states that if standardized controls that are used to monitor program changes, user access, and computer operations are effective, the auditor can conclude that the application controls are effective without having to repeat the previous year’s control test. This is referred to by PCAOB as benchmarking, and it is applicable if the auditor verifies that the application functionality control has not changed since the auditor last tested it.1

**User Acceptance Testing**

In addition to the static and dynamic code testing mentioned previously, software needs to be tested by the benefitting business units to ensure application functionality controls meet documented business rules, and the application interacts with input and output systems as intended. Issues identified in user acceptance testing can be categorized by the business owner as either:

1. Needing to be resolved prior to acceptance.

2. Authorized to be addressed in a subsequent release.

User acceptance testing controls are described in COBIT 2019 objective BAI07 Managed IT Change Acceptance and Transitioning, and in NIST SP 800-53r5 control SA-4 Acquisition Process. Business applications that are procured also go through user acceptance testing before being placed into service for the same reasons as developed software. Identified issues may need to be negotiated with the vendor to determine whether the delivered program meets contractual terms. Managing the documentation of requirements and plans for enhancements is an ongoing process for the benefitting business units and developers, whether in-house or external.

**Release Management and Software Escrow**

Code that has been tested and approved for use is compiled into an approved software version, which should be protected, for example, as described in NIST SP 800-53r5 controls SA-10 Developer Configuration Management and SC-34 Non-modifiable Executable Programs. Typically, a version control system is used to manage this process and enforce security objectives.

Approved versions should be stored offsite with a software escrow service to be used in the event that files or hardware are damaged or corrupted. An internal audit could verify whether management has ever tested the ability to recover operations from an escrowed version of production software.

**Security in Development**

In addition to the security-related steps mentioned previously, a vulnerability scan should be performed on an application after launching it into the production environment (but before opening it to full service) to identify configuration or component weaknesses. The results of the scan should be evaluated by IT-IS personnel and the business owner to determine whether the residual risk is acceptable. Controls over vulnerability scanning are described in COBIT 2019 objective DSS05 Managed Security Services; in NIST SP 800-53r5 control RA-5 Vulnerability Monitoring and Scanning; and in CIS Top 20 control 3 Continuous Vulnerability Management.

Vendor-provided software, including firmware in a business application ecosystem, is often updated to address security flaws in the code or interactions with component technologies in a new version called a patch. The controls over implementing patches are generally the same as for other new software versions, except there may be internal timeliness thresholds for patches that are not expected of other enhancements and the IS team typically has some responsibility for monitoring or enforcing those expectations. Patch management controls are covered in COBIT 2019 objectives BAI03 Managed Solutions Identification and Build and DSS05 Managed Security Services; in NIST SP 800-53r5 control SI-2 Flaw Remediation; and in CIS Top 20 control 3 Continuous Vulnerability Management.

**Production Support**

Controls to manage an in-service application include a number of significant IT-IS processes, including:
Hosting an approved version on a server and implementing new versions into production.

- Connecting the application to its databases.
- Connecting the system to internal or external interfaces, including application programming interface (API) middleware.
- Working with the benefitting business units to establish necessary system roles and authorization processes.
- Working with IS to implement encryption and authentication technologies.
- Monitoring system performance, and responding to errors and outages.
- Establishing system backup and recovery processes.

Some of these processes, such as system hosting, database administration, middleware management, encryption, and system backup and recovery are, or will be, covered in other GTAGs while certain aspects of identity and access management are covered in this Guide as well as in a separate GTAG. Still others, like configuration management and system performance monitoring, are primarily covered in this Guide, as they pertain to applications.

One aspect to consider in scoping a business application audit is whether controls over the application are implemented in standardized processes that are applied to other applications in the enterprise. As stated in the risk assessment table item #7, due to control inheritance, an audit of a business application may exclude from its scope any controls that are effected by standardized processes that are audited separately, although during planning, the audit team should verify the extent to which the business application is covered.

Configuration Management

One of the steps in designing a business application is establishing a baseline configuration, which documents the set of approved component technologies, interface settings, and other controls that make the application operational. The baseline configuration is often recorded in a service management application that helps coordinate and record changes, thereby updating the baseline configuration. Processing and output errors and other system performance issues are often caused or fixed by configuration changes.

Controls over configuration management and changes to in-service systems are described in greater detail in the GTAG IT Change Management; the COBIT 2019 domain Build, Acquire and Implement; the Change Management control family in NIST SP 800-53r5; and the CIS Top 20 control 5 Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers.

User Access Management

The GTAG Identity and Access Management discusses related controls at length, including the idea that if a business application is federated with standardized tools, then such controls may be scoped-out of an application audit. However, in certain higher-risk applications, an analysis of users with elevated privileges may be desired, even in federated applications, to determine whether users' supervisors are exercising perfunctory oversight. In the context of a business application, elevated privileges may mean higher financial approval thresholds or the ability to unmask...
protected data, whereas the term **privileged account** usually refers to system administrator, **super user** or **database administrator** roles.

If non-employees have access to a business application, there is a higher inherent risk if the application is not federated with the human resources information system for user IDs because the process is probably manual and reliant on vendor personnel to notify the system administrator when their personnel change roles or employment status.

System roles within business applications may be pre-defined and unalterable in some off-the-shelf or vendor-provided software but are more likely to be configurable by administrators to meet the needs of the benefitting business units. The roles, related permissions, and intended users should be documented and made available to user supervisors during authorization processes to help supervisors make informed authorization decisions.

Identity and access management controls are primarily described in: the COBIT domain Deliver, Service and Support; the NIST SP 800-53r5 control families Access Control, Identification and Authentication, and Personnel Security; and the CIS Top 20 control 14 Controlled Access Based on the Need to Know.

**Security in Production**

One of the (usually) configurable security-related controls in business applications is the recording of transactions and other events — like the creation of new roles or user accounts or the initiation and termination of user sessions — in log files. The NIST SP 800-53r5 control family Audit and Accountability provides excellent guidance for controls over logging. The IS and production support teams should work together to identify what types of events and information should be logged, implement agreed-upon solutions, and connect application logs to enterprisewide log monitoring tools. While logging event information provides data for other controls, the monitoring and analysis of event logs will be covered more extensively in other GTAGs primarily related to cybersecurity.

**Other Relevant Control Types**

Other control types that are relevant to, embedded in, or built on business application control processes include, but are not limited to, the ones described below.

**Records and Information Management**

The organization’s records and information management (RIM) program should recognize the following list as official records and establish requirements for retention:

- Application architecture diagrams.
- Data flow diagrams.
- Quality assurance and user acceptance testing routines and results.
- Source code for approved versions.
- Baseline configurations.
- System roles, and user account and permissions authorizations.
Event logs.

Controls over record retention are mainly described in COBIT 2019 objective BAI08 Managed Knowledge and in NIST SP 800-53r5 controls SI-12 Information Management and Retention and SA-5 System Documentation.

**Vendor Management**

Wherever external personnel or entities are involved in developing or supporting business applications, contracts should provide sufficient explanation of security and performance requirements. Contracts should include service level agreements and ongoing oversight, communication, and remediation processes defined, as appropriate. Controls over vendors are described in greater detail in the GTAG Information Technology Outsourcing; in COBIT 2019’s objectives APO08 Managed Relationships, APO09 Managed Service Agreements, and APO10 Managed Vendors; and in NIST SP 800-53r5 in several control families, primarily Program Management, Personnel Security, System and Services Acquisition, System and Communications Protection, and Supply Chain Risk Management.

**Asset Management**

Maintaining an inventory of business applications with sufficient metadata to support governance, security, and operational needs is a fundamental enabler of many IT-IS processes. Controls over a system inventory are described in COBIT 2019 objective APO01 Managed I&T Management Framework; NIST SP 800-53r5 control PM-5 System Inventory; and CIS Top 20 control 2 Inventory and Control of Software Assets. In the absence of a specifically-designated software inventory tool, the service management application may serve as a de facto inventory, with a good amount of configuration and management details, though it may not include all cloud-based applications or have all the desired metadata.

With vendor-provided applications, the unauthorized use of software licenses may also be a relevant concern. An audit could attempt to reconcile license assignees to the human resources information system to evaluate compliance with contract terms and cost management.

**Database Administration and Business Intelligence**

Application databases may store confidential information about transactions, customers, vendors, employees, or other sensitive data types critical to business operations. Many of the controls related to configuration management, identity and access management, and backup and recovery apply at the database layer, and personnel with database administrator roles may be in IT or the benefiting business units.

Database administration controls are often managed by enterprise-wide processes, so a business application audit might be primarily concerned with verifying the justifications for individual and system IDs — including APIs and middleware services — to have access to the records. Audits may also verify the use of encryption on tables or specific fields. Additionally, fraud detection and cybersecurity monitoring controls may be programmed to analyze actions taken in application databases.
Data governance and protection controls, which pertain more to the usage of data (sometimes referred to as business intelligence) than the administration of a database, are covered more extensively in other GTAGs.

Controls over database administration, data governance, and data protection are largely covered in COBIT 2019 objectives APO01 Managed I&T Management Framework and APO14 Managed Data; NIST SP 800-53r5 control families Change Management, Program Management, System and Communications Protection, and System and Information Integrity; and CIS Top 20 controls 14 Controlled Access Based on the Need to Know and 18 Application Software Security.

Using Computer-assisted Audit Techniques

Computer-assisted audit techniques (CAATs) can be found or created in a multitude of off-the-shelf applications and tools to enhance the breadth and efficiency of the audit process. The use of CAATs can enable a review of an entire population of transactions or records in a given period. Typical CAATs include reviews for duplicate vendors or transactions or separation of duty conflicts, while more advanced CAATs might look for other identity and access management anomalies. A well-designed audit supported by CAATs demonstrates adherence to Standard 1210 — Proficiency, which states in section 1210.A3 (emphasis added), “Internal auditors must have sufficient knowledge of key information technology risks and controls and available technology-based audit techniques to perform their assigned work.”

Conclusion

Business application controls cover the full life cycle and usage of the application, from planning and development to support and management reporting. When considering the risks and scope of a business application audit, it is important to understand how the application is used to support business needs and the role of vendors in every control category. This understanding can help the audit program focus on the highest risks in a way that provides value-added insight. Given the important role of applications as enablers of business processes, a risk-based audit plan should include audit engagements that evaluate standardized and system-specific controls to ensure significant risks are covered.
Appendix A. Relevant IIA Standards and Guidance

The following IIA resources were referenced directly or indirectly throughout this practice guide. For more information about applying the *International Standards for the Professional Practice of Internal Auditing*, please refer to The IIA’s [Implementation Guides](https://www.theiia.org).

<table>
<thead>
<tr>
<th><strong>Code of Ethics</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Principle 1: Integrity</td>
</tr>
<tr>
<td>Principle 2: Objectivity</td>
</tr>
<tr>
<td>Principle 3: Confidentiality</td>
</tr>
<tr>
<td>Principle 4: Competency</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Standards</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard 1210 – Proficiency</td>
</tr>
<tr>
<td>Standard 1220 – Due Professional Care</td>
</tr>
<tr>
<td>Standard 2110 – Governance</td>
</tr>
<tr>
<td>Standard 2130 – Control</td>
</tr>
<tr>
<td>Standard 2201 – Planning Considerations</td>
</tr>
<tr>
<td>Standard 2220 – Engagement Scope</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Guidance</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>GTAG: IT Essentials for Internal Auditors, 2020</td>
</tr>
<tr>
<td>GTAG IT Change Management, 2020</td>
</tr>
</tbody>
</table>
Appendix B. Glossary

Definitions of terms marked with an asterisk are taken from the “Glossary” of The IIA’s International Professional Practices Framework®, 2017 edition. Other definitions are either defined for the purposes of this document or derived from the following sources:


application – A computer program or set of programs that performs the processing of records for a specific function; Contrasts with systems programs, such as an operating system or network control program, and with utility programs [ISACA Glossary].

application functionality controls – The programmed routines and related parameters that enable software to execute according to business rules.

application programming interface (API) – A set of routines, protocols and tools referred to as “building blocks” used in business application software development. A good API makes it easier to develop a program by providing all the building blocks related to functional characteristics of an operating system. A programmer utilizes these APIs in developing applications that can operate effectively and efficiently on the platform chosen [ISACA Glossary].

application security – The set of system-specific and inherited IS controls applied to the development, operation and usage of an application.

architects – Personnel responsible for designing or approving systems that meet internal requirements and integrate with current or planned infrastructure.

asset management – A set of processes to record, safeguard and optimize the use of resources.

assurance services* - An objective examination of evidence for the purpose of providing an independent assessment on governance, risk management, and control processes for the organization. Examples may include financial, performance, compliance, system security, and due diligence engagements.

authentication – Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to resources in a system [NIST SP 800-53, Revision 5, Glossary].

authorization – Access privileges granted to a user, program, or process or the act of granting those privileges [NIST SP 800-53, Revision 5, Glossary].

availability – Ensuring timely and reliable access to and use of information. Ensuring timely and reliable access to and use of information [ISACA Glossary].

baseline configuration – An approved set of components, system settings and connections to other systems.
board* – The highest level governing body (e.g., a board of directors, a supervisory board, or a board of governors or trustees) charged with the responsibility to direct and/or oversee the organization’s activities and hold senior management accountable. Although governance arrangements vary among jurisdictions and sectors, typically the board includes members who are not part of management. If a board does not exist, the word “board” in the Standards refers to a group or person charged with governance of the organization. Furthermore, “board” in the Standards may refer to a committee or another body to which the governing body has delegated certain functions (e.g., an audit committee).

business intelligence – The use of data to present, analyze or predict business activities.

business owner – The leader of the business unit that receives the primary benefit from an IT resource. The business owner determines business requirements and authorizes acceptance of the resource. See also the definition of authorizing official in NIST SP 800-53r5.

business rules – Representations of business processes and constraints that are encoded into applications to fulfill user requirements.

compliance* – Adherence to policies, plans, procedures, laws, regulations, contracts, or other requirements.

component technology – A discrete technology asset that represents a building block of a system and may include hardware, software, or firmware. See also the definition of system component in NIST SP 800-53r5.

configure – Establishing the settings and connections necessary to make an application operational.

control* – Any action taken by management, the board, and other parties to manage risk and increase the likelihood that established objectives and goals will be achieved. Management plans, organizes, and directs the performance of sufficient action to provide reasonable assurance that objectives and goals will be achieved.

control inheritance – When a system or application receives protection from security or privacy controls (or portions of controls) that are developed, implemented, assessed, authorized, and monitored by entities other than those responsible for the system or application [NIST SP 800-53, Revision 5, Glossary].

control processes* – The policies, procedures (both manual and automated), and activities that are part of a control framework, designed and operated to ensure that risks are contained within the level that an organization is willing to accept.

customer relationship management – Software solutions that help an enterprise manage customer relationships in an organized manner [ISACA Glossary].

data input validation – Automated routine to check for meaningfulness of data input to a system, by comparing to a data dictionary or specified parameters.

database administrator (or administration) – An individual or department responsible for the security and information classification of the shared data stored on a database system. This
responsibility includes the design, definition and maintenance of the database [ISACA Glossary].

dynamic code testing – Analysis of software in operation, by using specified test routines and observing the results.

ecosystem – The hardware, firmware, software and connections that make up a business application’s environment.

encryption – The process of taking an unencrypted message (plaintext), applying a mathematical function to it (encryption algorithm with a key) and producing an encrypted message (ciphertext) [ISACA Glossary].

engagement* – A specific internal audit assignment, task, or review activity, such as an internal audit, control self-assessment review, fraud examination, or consultancy. An engagement may include multiple tasks or activities designed to accomplish a specific set of related objectives.

governance* – The combination of processes and structures implemented by the board to inform, direct, manage, and monitor the activities of the organization toward the achievement of its objectives.

identifier – A unique label used by a system to indicate a specific entity, object, or group [NIST SP 800-53, Revision 5, Glossary].
**industrial control system** – General term that encompasses several types of control systems, including supervisory control and data acquisition systems, distributed control systems, and other control system configurations such as programmable logic controllers found in the industrial sectors and critical infrastructures. An industrial control system consists of combinations of control components (like electrical, mechanical, hydraulic, and pneumatic) that act together to achieve an industrial objective (such as manufacturing, or the transportation of matter or energy) [NIST SP 800-53, Revision 5, Glossary].

**interface** – Common boundary between independent systems or modules where interactions take place [NIST SP 800-53, Revision 5, Glossary].

**log monitoring** – using specialized software to scan event logs for patterns or anomalies that may indicate unauthorized accounts, access or activities.

**metadata** – Information that describes the characteristics of data, including data format, syntax, semantics, and contents [NIST SP 800-53, Revision 5, Glossary].

**middleware** – Another term for an API, it refers to the interfaces that allow programmers to access lower- or higher-level services by providing an intermediary layer that includes function calls to the services [ISACA Glossary].

**patch** – Fixes to software programming errors and vulnerabilities [ISACA Glossary].

**persistent** – A characteristic of stored data that keeps it the same, enabling later retrieval.

**point-of-sale system** – Enables the capture of data at the time and place of transaction; such terminals may include use of optical scanners for use with bar codes, or magnetic card readers for use with credit cards. Point-of-sale systems may be connected online to a central computer, or used as stand-alone terminals that hold the transactions until the end of a specified period, then sending data to the main computer for batch processing [ISACA Glossary].

**privileged account** – A user that is authorized (and therefore, trusted) to perform security-relevant functions that ordinary users are not authorized to perform [NIST SP 800-53, Revision 5, Glossary].

**production support** – Processes to configure, administer and troubleshoot applications. See also ISACA’s definition for IT service.

**records and information management** – An enterprisewide program to identify official record types and their storage locations, and establish retention and destruction requirements.

**risk** – The possibility of an event occurring that will have an impact on the achievement of objectives. Risk is measured in terms of impact and likelihood.

**security categorization** – The characterization of information or an information system based on an assessment of the potential impact that a loss of confidentiality, integrity, or availability of such information or information system would have [NIST SP 800-53, Revision 5, Glossary].
**separation of duties** – A basic internal control that prevents or detects errors and irregularities by assigning to separate individuals the responsibility for initiating and recording transactions and for the custody of assets [ISACA Glossary].

**shadow IT** – Personnel or resources performing an IT function outside of the IT management hierarchy.

**software (or system) development life cycle (SDLC)** – The phases deployed in the development or acquisition of a software system. Typical phases of SDLC include the feasibility study, requirements study, requirements definition, detailed design, programming, testing, installation and post-implementation review, but not the service delivery or benefits realization activities [ISACA Glossary].

**source code** – The language in which a program is written. Source code is translated into object code by assemblers and compilers [ISACA Glossary].

**Standard** – A professional pronouncement promulgated by the International Internal Audit Standards Board that delineates the requirements for performing a broad range of internal audit activities and for evaluating internal audit performance.

**static code testing** – An automated analysis of code to detect potential errors, vulnerabilities, or inefficient coding.

**super user** – A type of system admin role that has all permissions, including root access to the operating system.

**system administrators** – Personnel authorized to configure and support the operation of an IT resource.

**system roles** – Sets of permissions within an application that typically correspond to job functions.

**technology planning** – Activities to align IT-IS resources with business needs, ensuring objectives of confidentiality, integrity, availability, privacy and security are met. See also ISACA’s definition for Strategic Planning, and NIST SP 800-53r5’s definition of Enterprise Architecture.

**technology roadmap** – A plan for a business application’s version and component updates, aligned with the enterprise architecture plan. See also ISACA’s definitions for IT strategic plan and IT tactical plan.

**vendor management** – A set of processes to procure goods and services, ensure acceptable delivery or performance, and resolve disputes.

**version control system** – An application used in SDLC to manage changes to the source code and facilitate approvals to promote code from the development environment to the test environment, and then to the production environment.

**vulnerability scan** – Automated routine to detect known weaknesses in software code or configurations. The vulnerabilities may be assigned a score to facilitate prioritization of resolution efforts.
**web application firewall** – A firewall placed between the internet and an application server to filter traffic and prevent various types of cyberattacks.
Appendix C. References


Acknowledgements

Guidance Development Team

Susan Haseley, CIA, United States (Chairman)
Justin Pawlowski, CIA, CRMA, Germany (Team Lead)
Brad Ames, United States
Jim Enstrom, CIA, United States
Ruth Mueni Kioko, Kenya
Mike Lynn, CIA, CRMA, United States
Sajay Rai, CISM, CISSP, CISM, United States
Terence Washington, CIA, CRMA, United States

IIA Global Standards and Guidance

David Petrisky, CIA, CRMA, CISA, CPA, Director, (Project Lead)
Jim Pelletier, Vice President
Harold Silverman, Managing Director
Laurella Nelson, Senior Editor
Christine Janesko, Content Writer and Technical Editor
Geoffrey Nordhoff, Content Writer and Technical Editor

The IIA thanks the following oversight bodies for their support: Information Technology Guidance Committee, Professional Guidance Advisory Council, International Internal Audit Standards Board, Professional Responsibility and Ethics Committee, and the International Professional Practices Framework Oversight Council.

Commented [DP8]: Make sure spacing is consistent.

Commented [DP9]: Delete? The Committee no longer exists.
About The IIA
The Institute of Internal Auditors (IIA) is the internal audit profession’s most widely recognized advocate, educator, and provider of standards, guidance, and certifications. Established in 1941, The IIA today serves more than 200,000 members from more than 170 countries and territories. The association’s global headquarters is in Lake Mary, Fla., USA. For more information, visit www.globaliia.org.

Disclaimer
The IIA publishes this document for informational and educational purposes. This material is not intended to provide definitive answers to specific individual circumstances and as such is only intended to be used as a guide. The IIA recommends seeking independent expert advice relating directly to any specific situation. The IIA accepts no responsibility for anyone placing sole reliance on this material.

Copyright
Copyright © 2020 The Institute of Internal Auditors, Inc. All rights reserved. For permission to reproduce, please contact copyright@theiia.org.

June 2021