



Integrating Security and IT

Julia H. Allen

October 2006

ABSTRACT: This article describes the key relationship between IT processes and security controls. Most, if not all, IT processes play a critical role in sustaining a desired security state during deployment and operations.

INTRODUCTION

As time passes and system complexity increases, a risk-centered practice approach applied at a local level often is not sufficient to sustain an adequate level of security or software assurance. In response, some organizations¹ have found that adding an IT process-centered practice approach to the approaches described in Plan, Do, Check, Act and Risk-Centered Practices aids in being able to measure and sustain adequate security.

This article describes

- two widely adopted frameworks, ITIL® and COBIT®, that should be considered when determining how best to embed security controls into defined IT operational processes. These frameworks describe sound processes, practices, and control objectives for managing and operating IT systems, including their security state. Organizations using these frameworks report an increased ability to deliver high quality service to their customers, which includes being able to measure and satisfy confidentiality, availability, and integrity requirements.
- an approach to embedding security controls in IT processes based on work done by the IT Process Institute.

Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-2612

Phone: 412-268-5800
Toll-free: 1-888-201-4479

www.sei.cmu.edu

¹ Refer to [Stern 01] and [Worthen 05] for several examples.

IT SERVICE MANAGEMENT: ITIL²

The benefit of a process-centered approach for developing software is well documented and used in segments of the U.S. and international software development communities (refer to the SDLC Process content area). Similarly, use of a process-centered approach for deploying and operating IT-based systems is also well documented [OGC 07] and used particularly in Europe and the United Kingdom.

ITIL Processes

The current version of ITIL (version 3) consists of a set of 5 core publications, each defined as a set of processes and functions that include high-level overviews as well as detailed definitions of the steps in each process. (Refer to Navigating the Security Practice Landscape for further details.)

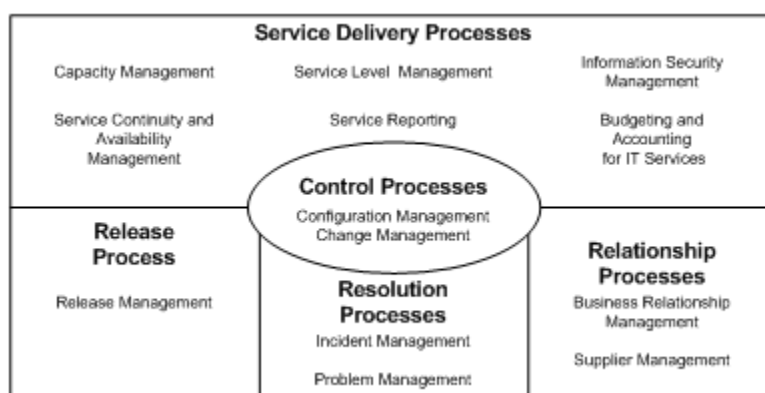


Figure 1. Service management process [ISO 05c]

Here is a brief description of the most relevant processes depicted in Figure 1 and their relationship to ensuring adequate security during deployment and operations. These are excerpted and summarized from [ITIL 00], [ITIL 01], and [itSMF 07].

Capacity Management

This process is responsible for ensuring adequate capacity is available at all times to meet business requirements. Capacity Management is involved in eval-

² See also the Information Technology Infrastructure Library wikipedia entry [ITIL 08], <http://www.best-management-practice.com/Knowledge-Centre/Best-Practice-Guidance/ITIL/>, and <http://www.itil-officialsite.com/home/home.asp>.

uating all changes and the cumulative effect of changes over time. Aspects of Capacity Management address

- throughput capacity,
- response times,
- utilization of each component of the IT infrastructure,
- capacity demands,
- workload allocation to IT system components,
- resources required for each application, and
- modeling of all of these.

Many of the aspects are affected and provide evidence when a security breach occurs. Monitoring these aspects aids in intrusion prevention and detection.

Service Level Management

This process is responsible for ensuring service level agreements (SLAs) are met (including security requirements such as availability and compliance with security standards). This process involves assessing the impact of any change on service quality and SLAs, both when changes are proposed and after they have been implemented. This includes security patches, changes to security configurations, and updated and new security technologies.

IT Service Continuity Management

This process is responsible for managing an organization's ability to continue to provide a predetermined and agreed-on level of IT service to support minimum business requirements following an interruption to the business. IT service continuity can be greatly affected by a successful security breach. Continuity actions are required to plan for, prevent, detect, and respond to a breach and restore service as quickly and reliably as possible.

Availability Management

This process is concerned with the design, implementation, measurement, and management of IT services to meet business availability requirements. This includes an understanding of why IT system failures occur (including those caused by security events) and the time to resume service. Availability depends on reli-

ability, maintainability, serviceability, and resilience,³ all of which can be affected by unauthorized access and other forms of security compromises.

Release and Deployment Management

This process is responsible for ensuring a secure, managed rollout of new and updated software and system components. This includes assurance that

- the correct software is included,
- it has been tested in advance,
- it includes only authorized changes,
- it is free of malicious code and remains so during distribution and deployment, and
- new releases can be backed out to a prior known and trusted state.

Incident Management

ITIL defines an incident as “an unplanned interruption to an IT service, or a reduction in the quality of an IT service. Failure of a configuration item that has not yet impacted service is also an incident.” [itSMF 07]. Security events fall under this definition. This process relies on the service desk function, which

- logs, categorizes, and prioritizes all incidents
- performs first-line investigation and diagnosis
- manages the life cycle of incidents, escalating as appropriate and closing them when the user is satisfied
- keeps the user informed of incident status

This process interfaces with Problem Management to ensure adequate resolution of all incidents.

Problem Management

This process encompasses the capture, recording, and analysis of all software and system problems (also referred to as incidents). It attempts to discover the links between incidents so that the cause(s) can be determined. This includes

³ The reliability of an IT service indicates the degree to which the service offers the agreed functionality during an indicated period of time. Maintainability is an indication of the ease with which maintenance can be carried out on a service. Serviceability means the way contracts with third parties are dealt with. Resilience is the ability of an IT service to continue to operate properly, in spite of the malfunctioning of one or more subsystems [ITIL 99].

taking improvement actions to prevent known incidents from recurring. Problem Management produces change requests that serve as input to Change Management.

Service Asset and Configuration Management

This process

- is responsible for producing and maintaining current, accurate, and comprehensive information about, and inventories of, all components (service assets and configuration items) of the IT infrastructure
- ensures that only authorized components are included in any production configuration
- identifies relationships between any system component that is to be changed and any other components with which it interfaces or for which there is some interdependency that could be affected by the change. In this way, the process allows all component owners to be involved in the change impact assessment process.
- incorporates asset management, where the assets being managed are components of the IT infrastructure (hardware, software, data, information, etc.). Asset management includes asset classification as described in Risk-Centered Practices and FIPS 199 Security Categorization of Federal Information and Information Systems [NIST 04].

Change Management

The goal of this process is to control and manage all changes to any component of the IT system. This includes security patches, updates, and new technologies and tools. The output of this process is an evaluated and authorized change. The evaluation of any change includes determining its impact on the current security posture of the system, including, where possible, anticipating new problems or incidents that may be introduced by the change.

Change Management is one of the most, if not the most, critical processes with respect to deploying and operating secure systems and software. The Change Advisory Board (or equivalent) that reviews all changes has the responsibility for ensuring that the level of security is not reduced below an acceptable level by a change and that this has been adequately demonstrated in pre-production testing.

CONTROL OBJECTIVES FOR IT: COBIT®

COBIT (Control Objectives for Information and related Technologies) [ITGI 07a] describes a framework for IT governance and management. It is intended to ensure the effectiveness, efficiency, confidentiality, integrity, availability, compliance, and reliability of information and the systems used to process this information.

According to Wikipedia:

The COBIT mission is “to research, develop, publicize and promote an authoritative, up-to-date, international set of generally accepted information technology control objectives for day-to-day use by business managers and auditors.” Managers, auditors, and users benefit from the development of COBIT because it helps them understand their IT systems and decide the level of security and control that is necessary to protect their companies’ assets through the development of an IT governance model.

COBIT Framework

The COBIT Framework is organized into four domains, 34 high-level control objectives, and 318 detailed control objectives. The framework follows a general Plan-Do-Check-Act structure. The domains and control objectives are as follows:

Plan and Organize

- P01 Define a strategic IT plan.
- P02 Define the information architecture.
- P03 Determine technological direction.
- P04 Define the IT processes, organization, and relationships.
- P05 Manage the IT investment.
- P06 Communicate management aims and direction.
- P07 Manage IT human resources.
- P08 Manage quality.
- P09 Assess and manage IT risks.
- P10 Manage projects

Acquire and Implement

- AI1 Identify automated solutions.
- AI2 Acquire and maintain application software.
- AI3 Acquire and maintain technology infrastructure.
- AI4 Enable operation and use.
- AI5 Procure IT resources.

- AI6 Manage changes.
- AI7 Install and accredit solutions and changes.

Deliver and Support

- DS1 Define and manage service levels.
- DS2 Manage third-party services.⁴
- DS3 Manage performance and capacity.
- DS4 Ensure continuous service.
- DS5 Ensure systems security.
- DS6 Identify and allocate costs.
- DS7 Educate and train users.
- DS8 Manage service desk and incidents.
- DS9 Manage the configuration.
- DS10 Manage problems.
- DS11 Manage data.
- DS12 Manage the physical environment.
- DS13 Manage operations.

Monitor and Evaluate

- ME1 Monitor and evaluate IT performance.
- ME2 Monitor and evaluate internal control.
- ME3 Ensure regulatory compliance.
- ME4 Provide IT governance.

COBIT and Security

As for ITIL, the scope of COBIT is all of IT. And as for ITIL processes, selected COBIT control objectives can be interpreted and meaningfully applied to system and software security. Such relationships are clear with a detailed review of the COBIT documentation.

Example

As one example, the DS5 Delivery and Support, Ensure Systems Security high-level control objective includes 11 **detailed control objectives**:

⁴ See also Outsourcing Managed Security Services [Allen 03] for a description of practices to consider when outsourcing security services for an operational system.

- DS5.1 Management of IT security
- DS5.2 IT security plan
- DS5.3 Identity management
- DS5.4 User account management
- DS5.5 Security testing, surveillance, and monitoring
- DS5.6 Security incident definition
- DS5.7 Protection of security technology
- DS5.8 Cryptographic key management
- DS5.9 Malicious software prevention, detection, and correction
- DS5.10 Network security
- DS5.11 Exchange of sensitive data

ITGI Guide

ITGI has published a useful guide titled COBIT Security Baseline: An Information Security Survival Kit [ITGI 07b]. This guide identifies all of the control objectives from COBIT version 4.1 that need to be considered for security, actions to take, traceability to the prior version of ISO 17799 [now ISO 05a], and “survival kits” of control objectives for home users, professional users, managers, executives, senior executives, and board of directors/trustees.

EMBEDDING SECURITY CONTROLS IN IT PROCESSES

Work performed in collaboration with the IT Process Institute and the Institute of Internal Auditors has provided access to a community of practitioners who operate large, complex, highly secure, highly available operational systems. Practitioners responsible for deploying and operating such systems accomplish this, in part, by embedding well-defined security controls into mature IT operational processes such as change management, configuration management, and release management (using, for example, ITIL process definitions).⁵

Change management is an example of an IT operational process that is critical for adequate security because of its relationship to availability and security patch management. “Gartner research shows that an average of 80 percent of mission-

⁵ Practices from this community are described in The Visible Ops Handbook [ITPI 04], in “High Performing IT Organizations: What You Need to Change to Become One” [Kim 04], and in Change and Patch Management Controls: Critical For Organizational Success [IIA 05].

critical application service downtime is directly caused by people or process failures. The most common cause of people and process failures is change” [Scott 01].

From an IT process-centered perspective, every security patch should be managed and deployed using a rigorous change management process that evaluates system risks that may be introduced by implementing (or not implementing) the change. To effectively sustain adequate security with respect to patch management, the process used should be an instantiation of the same one used for other aspects of change management during the software development life cycle.

Visible Ops and Visible Ops Security

The Visible Ops method [ITPI 04] describes this concept in detail, identifying the steps necessary to get an IT infrastructure that is out of control under control. This method takes into account technical, management, performance, monitoring, and audit practices for both operations and security.

Visible Ops derives from extensive field work and experience (conducted since 2000) with organizations responsible for predominantly IT-based operational systems. The organizations from which this method derives represent “high performing IT operations and security organizations with the highest service levels, as measured by

- mean time to repair (MTTR)
- mean time between failures (MTBF)
- the early integration of security requirements into the operations lifecycle
- the lowest amount of unplanned work (unexpected work created by the installation of a change)
- the highest server to system-administrator ratios”⁶ [ITPI 04]

These qualitative measures were documented in The Visible Ops Handbook in 2004. They were quantitatively measured and validated in the IT Controls Performance Study in 2006 [Kim 06]. The results are discussed in Prioritizing IT Controls for Effective, Measurable Security. The performance gap between high and low performers was confirmed, indicating that high performers had

⁶ Greater than 100 servers for each system administrator as compared to, on average, 15 servers for each administrator in organizations not using an IT process-centered approach.

measures that were typically 5 to 8 times better than the medium and low performers.

The Visible Ops method consists of the four phases briefly described below.

Phase 1 – Stabilize the Patient and Modify First Response

Practices in this phase are intended to reduce the number of system outages by prohibiting changes outside of scheduled maintenance windows. This phase helps problem managers perform their first-responder duties by providing change-related information so they can determine the cause of an outage. One of the key concepts when a problem occurs is to determine who made the last configuration change and when it was made, rather than simply logging into the infrastructure and starting to troubleshoot.

Phase 2 – Catch and Release, and Find Fragile Artifacts

Practices in this step include creating an inventory of assets, configurations, and services to identify those that are most fragile. Fragile artifacts are those that

- are highest risk
- require the most maintenance
- have the lowest change success rates (introducing a change produces additional problems).
- have the highest mean-time-to-respond (MTTR)
- have the highest downtime costs for the system and the organization

Fragile artifacts are given extra attention to avert high-risk changes and additional unplanned work.

Phase 3 – Establish a Repeatable Build Library

The highest return on investment comes from implementing effective release management processes. This step creates repeatable builds for the most critical assets and services to make it more cost effective to rebuild than to repair. Addressing security as part of preproduction processes can have a significant positive effect. For example, security controls to ensure that all builds are hardened and that critical configuration settings are set properly are implemented during this phase. These controls aid in establishing a defensible production posture that significantly eliminates reactive production work downstream.

Phase 4 – Enable Continuous Improvement

The previous steps have progressively built a closed loop between the release, control, and resolution processes (see Figure 1). This step implements metrics to

enable the continuous improvement of all of these processes to best meet organizational and system objectives.

The Visible Ops concept has been expanded to address information and software security in greater depth in an IT Process Institute guide titled Visible Ops Security: Achieving Common Security and IT Operations Objectives in 4 Practical Steps [ITPI 08]. This work is based on additional 2007 research with the community of high-performing organizations described in the IT Controls Performance Study [Kim 06].

The Visible Ops Security method consists of the four phases briefly described below [ITPI 08].

Phase 1 – Stabilize the Patient and Get Plugged Into Production

Practices in this phase include gaining situational awareness and working to integrate security into daily IT operational processes such as change management. Security identifies where access needs to be more effectively controlled and reduced. Security develops security incident handling procedures that are integrated with the IT operations incident management process. This first phase results in quicker detection and correction of security incidents, reduces their likelihood, and helps IT operations increase availability and reduce unplanned work.

Phase 2 – Find Business Risks and Fix Fragile Artifacts

During this phase, security actively participates in a top-down, risk-based assessment and management approach (see Risk-Centered Practices) to identify business-critical assets that require the greatest level of protection. Key IT services and systems are identified, along with determining what IT controls are required to protect critical functionality. IT controls are streamlined for regulatory compliance and control issues are identified and corrected.

Phase 3 – Implement Development and Release Controls

The objective of this phase is to improve the quality of software releases to ensure that security standards are integrated into projects and software builds. Security integrates with internal audit, project management, the software development life cycle, and release management. Security works with accounting and purchasing (acquisition) to increase situational awareness regarding the security-readiness of acquired software and services.

Phase 4 – Continual Improvement

Security selects and implements relevant short-term and long-term effectiveness measures and ensures these are incorporated into organizational process improvement efforts.

CONCLUSION

While an IT process (ITIL) or control objectives (COBIT) view may not be applicable to all categories of software-intensive systems, they are useful models that should be considered when determining how best to embed security controls into defined IT operational processes. Most, if not all, of the IT processes identified in Figure 1 and described in COBIT play a significant role in achieving and sustaining an adequate level of security in an operational system.⁷

Sustainability is likely to be one of the key outcomes of including well-defined security controls as part of mature IT operational processes. If IT processes are essential to successful system operation and security controls are integrated with IT process definitions (and implementations), the organization is in a better position to deploy and operate a system that is sufficiently secure for the system's and organization's mission.

⁷ The IT Governance Institute and Office of Government Commerce report "Aligning COBIT® 4.1, ITIL® V3, and ISO/IEC 27002 for Business Benefit" [ITGI 08] provides useful guidance on how to integrate these three standards.

Copyright © Carnegie Mellon University 2005-2012.

This material is based upon work funded and supported by Department of Homeland Security under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center sponsored by the United States Department of Defense.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of Department of Homeland Security or the United States Department of Defense.

References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by Carnegie Mellon University or its Software Engineering Institute.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

This material has been approved for public release and unlimited distribution except as restricted below.

Internal use:* Permission to reproduce this material and to prepare derivative works from this material for internal use is granted, provided the copyright and “No Warranty” statements are included with all reproductions and derivative works.

External use:* This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other external and/or commercial use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

* These restrictions do not apply to U.S. government entities.

DM-0001120