SAGE
THE SYSTEM ADMINISTRATORS GUILD

ISBN 1-880446-21-9

6 Geoff Halprin A System Administrator's Guide to Auditing SAGE

6 Short Topics in Systems Administration

Edited by William LeFebvre

A System Administrator's Guide to Auditing

Geoff Halprin

SAGE
THE SYSTEM ADMINISTRATORS GUILD

6 *Short Topics in*
Systems Administration

Edited by William LeFebvre

A System Administrator's Guide to Auditing

Geoff Halprin

Published by the USENIX Association for
SAGE, the System Administrators Guild
2000

©Copyright 2000 by Geoff Halprin
ISBN 1-880446-21-9

To purchase additional copies and for membership information, contact:

The USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA USA 94710
Email: office@sage.org
Web: http://www.sage.org

First Printing July 2000

USENIX and SAGE are registered trademarks of the USENIX Association.
USENIX acknowledges all trademarks herein.

Printed in the United States of America, on 50% recycled paper, 10–15% post-consumer waste.

Contents

Foreword

Few things will ruin a system administrator's day faster than the announcement of an audit. It sometimes seems this practice is arbitrarily invoked by higher management for the sole purpose of adding to the workload of an already overworked support staff. Many view the auditing process as an insult, or an indication that someone is questioning the team's abilities or efforts.

The truth is that an audit is nothing more than a tool. Like any tool it can be effective or it can be abused. This booklet demystifies the process of an audit, revealing it for what it is. It can also help to turn the tables by revealing an audit not as something to dread, but as something to use in a way that has direct benefit to our jobs as systems administrators.

William LeFebvre
Alpharetta, GA, May 2000

Glossary

Audit Sponsor: The person who's paying for the audit to be performed. Whilst there are a number of stakeholders, there is usually only one sponsor. It is important to satisfy as many stakeholders as you can. It is essential to satisfy the sponsor.

Body of Knowledge: The definition of the problem space, and the associated breakdown of the problem space into a hierarchical structure. The BOK is used as the basis for the audit process.

System under Scrutiny: The system being studied as the subject of the audit.

Time Window: The period of time over which the audit, or certain aspects of it, takes place (e.g., the system probe window).

Preface

Auditing is not, at first appearance, an exciting topic. So why would anyone volunteer to spend months of effort writing a booklet that may never be read? Well, for a real answer, we have to look beyond the simple, well-known notion of the security audit.

Auditing is about the rigorous examination of a system, the identification of any shortfalls in compliance or practices, and the organised repair and improvement of that system. It is about ensuring that appropriate controls and processes are in place, and that staff are able to perform their jobs with appropriate training, resources and support.

Auditing is a very important tool for technical communities such as system administrators to have at their disposal. It can help them gain a better appreciation of the surrounding context of their actions and the impact of the changes they are planning. It allows them to step out of the day-to-day mire of problems, to see the wood for the trees, and to plot the path forward.

It is also a system administrator's most valuable tool in convincing management that there is a problem that needs addressing or a subsystem that needs improvement. It is your most strategic way to show management the true nature of what it is you do for a living, and how much effort is involved in doing that job well.

Most important, auditing should be *your* tool. It should not be seen as some bi-annual chore inflicted upon you from above. It should be the tool you call on to help you in understanding complex systems and in implementing change, including enlisting necessary support from management.

A Personal Perspective

I have been a consulting system administrator for over ten years. In that time, I have moved from site to site, cleaning up the mess of generations of decay. Each time I encountered the same problem: how do I determine what sort of mess I'm dealing with, and how do I fix that mess? Like many of my peers, as my first step I built up a toolkit of scripts and utilities of my own and from the Internet to help me.

Systems administration is a very young profession. Each of us builds up our own toolkits and procedures, and this leads to the great hidden cost of staff turnover; every system administrator re-creates each site they administer in their own image—one that they are comfortable with. This is a major source of the very entropy that we seek to conquer. It is clear that we, as a profession, need to work towards a common, well-defined framework and set of standards of practice. But that's another story.

As a consultant newly encountering a site with a prior history of consultant-instigated changes, I had a major task ahead of me of a non-technical nature: I had to convince management that the changes I was proposing were necessary. Actually, that's not quite correct. They were usually already feeling the effect of decay on their operations in the form of reduced system and application availability. The problem was really to convince them that there was a solid basis for believing that the changes *I* was proposing would actually improve things. (After a couple of generations of consultants making such promises, people become a little suspicious of such claims.)

What I discovered was that the audit, if done correctly, is a system administrator's most valuable tool in dealing with management. Indeed, the only reason I initially undertook a formal audit was for management's sake. Since then, I have performed a variety of audits of varying complexity and breadth of coverage. Each one of those has been vital to obtaining organisational commitment to change.

Over time, my original questions have become slightly more refined: how do I quantify the degree of entropy, and how much effort is required to repair the practices of the site? These are subtly different questions.

The audit is the starting point for technical improvement works. It quantifies how much effort (and money) is going to be required to improve things, raises management's awareness of the complexity of the systems administration role, and allows both the system administrator and management to review progress against agreed goals.

Goals of This Booklet

As should become clear, this is not another text on auditing computers by an auditor. Nor is it intended to replace the use of properly trained auditors. This booklet is directed towards the systems administration community, and it focuses on helping to bridge the gaps in understanding between that community and those with whom it interacts.

Reading this booklet will not make you an auditor. Those interested in becoming an auditor should consult one of the resources mentioned in Appendix B.

It should, however, help you and professionals from similar technical communities to conduct audits of varying degrees of formality, and to understand and use formal techniques to assist you in your role. It should especially help you in proactively planning your workload and in communicating with other key business communities. It should also provide a better understanding of the role and importance of auditing and, hence, better equip you to facilitate any external auditor so that both you and your organisation reap the maximum benefit.

The audit is your friend. If done right, it will get you a direct line to your manager's ear, with positive effects throughout the organisation.

My goal in writing this booklet is to help you, the system administrator, to harness the power that an audit provides.

Acknowledgments

I would like to thank a number of people for their help in bringing this text to fruition. Thanks go to Hal Miller, who forced me to write this thing in the first place.

Thanks also to Anton Aylward, Mark Teicher, Bret Watson, Richard Dempsey, Gordon Rowell and Mike Ciavarella for their comments on the text at various stages.

Finally, a special thanks to Bill LeFebvre for his work in shepherding this text through the editorial process, and for pushing me to completion, if on a somewhat expanded timeframe than originally planned.

An Apology

My apologies to any non-U.S. readers. This booklet was written in English. Any Americanisation of the text occurred during the editing stage.

1. Introduction

Why Audit?

There are three basic technical reasons for performing an audit:

- **To Control.** To gain control of system resources unknown to the auditor (or audit sponsor).

  This can be thought of as a *familiarisation* audit. Here we are measuring a system against independent criteria, in an attempt to gain an understanding of the system under scrutiny.

- **To Verify.** To verify the operation of a known system against established specifications or baselines.

  This is also known as a *compliance* audit. A baseline measure of the compliance of the system against a standard has already established, and we are seeking to re-measure the system against that same standard, noting any changes in the level of compliance.

- **To Measure.** To measure the impact of a planned revision or change to a known system.

  This can be thought of as a *progress* audit. Here we audit a system prior to implementing a change to that system, establishing a baseline against which we can measure the impact of the change itself.

Beyond these technical reasons, we find that the audit also serves other valuable purposes:

- **To Educate**

  The audit report represents a comprehensive, structured sweep through a problem space. This then serves as a primary source of education to others as to the complexity of that problem space, and the difficulties in addressing any shortfalls uncovered.

  I have often heard the complaint that management do not understand what system administrators do. It is, of course, the responsibility of the systems administration community (as the ones who do understand) to remedy this. The audit can help address the issue of education directly.

- **To Justify**

  It is all well and good to "know" how to fix the mess you have, but this intuition (no matter how much experience is supporting it) will not generally convince management that they should invest in such improvement works. Management needs to know what benefit they are likely to see, and how likely they are to see it.

The audit report shows (1) a clear understanding of the problem space, (2) specific problems with the present situation, and (3) quantifiable steps to be taken to improve relevant practices.

- **To Facilitate**

  By providing a comprehensive report on the problem space, with a list of recommended actions, you are most of the way to formulating a project plan for technical improvement works. Thus, the audit enables the systems administration team to gain control over the system in question, and to plan its repair and improvement.

Many auditors tend to focus on the technical aspects of the audit's value. By doing so, they do themselves and their clients a disservice. As this booklet will (I hope) show, the audit's greatest value is as a tool for communication and education. This is a point that I will reiterate throughout this text.

Three Audit Perspectives

One of the drivers behind writing this booklet was to help system administrators understand more about the people who benefit from audits, and what each of these classes of reader is looking for.

As a system administrator, I used technical audits as a way to work out what was broken and what wasn't. The resulting audit was useful to me, but it was not very useful to management. Nor would either management or the audit department have approved it as a "real" audit. This booklet is intended to help bridge the understanding gaps between the three main groups of people who must participate in an audit and understand its results, if those findings are to be of most value to the organisation: the auditor, management, and the systems administration staff.

The System Administrator's Perspective

The system administrator's perspective is a very down-to-earth, technical one. They want to know how to get their job done. Their questions are simple:

- What sort of mess am I dealing with?
- How do I fix it?
- How do I justify this effort to management?

An audit and its associated report provide an effective, structured way to answer these three questions.

By performing an audit, rather than random repairs, you are forced to complete an orderly examination and review of all aspects of the system under scrutiny. By separating out the phases of *assessment* and *repair*, you are better placed to plan your approach and, hence, estimate the effort and other costs involved in that repair.

A well-written audit report will not only serve as a precursor to a full project plan, but will help educate management and other readers as to the nature and complexity of the role of systems administration, and the true difficulties in effecting repairs to a production computing environment.

Management's Perspective

The role of management is to predict and control expenditure and to make recommendations and decisions based upon cost-benefit analyses presented to them by their staff. In terms of managing computer operations, the two basic questions that a manager is most often seeking to answer are:

- What sort of mess am I dealing with?
- How much effort (people, time, and money) is required to fix it?

As a system administrator, it is your job to answer these questions, and you must be prepared to provide some level of assurance that, if money is spent and effort is expended as per your recommendations, the problems will indeed be solved (or at least objectives met), and quantifiable benefits will accrue to the organisation.

Employees cannot really be held to account for their recommendations, beyond being fired. This threat may not provide sufficient assurance of a good result to a company that is about to expend tens or hundreds of thousands of dollars, based solely on your intuition. It is important, therefore, to provide adequate justification as to why your recommendations are sufficient, if followed, to provide the intended outcome. The audit report is the principal mechanism for such a justification.

The Auditor's Perspective

Finally, we have the professional auditor’s perspective. The auditor brings to the table a formality of process and objective. The true underlying nature of an audit is to measure compliance against some pre-defined standard. The definition of that standard is a separate exercise. Auditors understand the process by which such an assessment of compliance should be performed, and the ground rules for performing that assessment.

Typical questions an auditor seeks to answer include:

- What are the standards against which we are measuring compliance?
- What is the methodology by which we are to measure that compliance?
- Is there access to all of the information necessary to conduct the audit?
- Has an accurate baseline measurement been established prior to this audit?
- Have accurate measurements been taken during the present audit?

Applying these generic rules of process and methodology to the continually changing world of technology is a difficult task. Whilst all auditors should have the ability to apply set rules and perform set audits, they may not have the requisite depth and breadth of technical experience nor the solid foundation in the underlying concepts of operating systems, information security, and industry best practices that are required to apply these principles within the highly individualised world of computer operations.

Having said that, however, I believe it is therefore vital that technicians who are intending to perform audits make the effort to understand the principles and practices of the audit profession, such that any audit you perform can withstand formal scrutiny.

Even if it is never your intention to perform a formal audit, the concepts presented in this booklet will help you to do your job in a manner that is more proactive, controlled, and beneficial to all involved.

Auditing as an Agent for Positive Change

An audit is about more than just a review of process, controls, mechanisms, and audit trails. It is about reviewing policy, ensuring that adequate staffing levels and effective training programmes are in place, demonstrating quality control, and measuring and demonstrating progress to management and others. It is also about ensuring that appropriate resources, budget, and profile are given to particular systems, based upon management's commitment to those systems.

The nature of an audit is that it seeks to study the system in depth, providing a mechanism both for capturing and describing the complexity of that system, and for defining what support is necessary to implement and maintain that system successfully. This, in turn, presents the necessary organisational impetus for change.

The audit, and in particular the audit report, can help to break down communication barriers between the various communities, and to align their efforts and energies.

2. What Is an Audit?

> *audit* 1. An official examination and verification of accounts and records, esp. of financial accounts. (*Macquarie Dictionary*, 2nd rev. ed.)
>
> 2. The systematic examination of records and documents and the securing of other evidence by confirmation, physical inspection, or otherwise, for one or more of the following purposes: determining the propriety or legality of proposed or consummated transactions, ascertaining whether all transactions have been recorded and are reflected accurately in accounts; determining the existence of recorded assets and inclusiveness of recorded liabilities; determining the accuracy of financial or statistical statements or reports and the fairness of the facts they present; determining the degree of compliance with established policies and procedures relative to financial transactions and business management; and appraising an accounting system and making recommendations concerning it. (*The 'Lectric Law Library's Reference Room*)
>
> 3. A methodical examination and review. (*Merriam Webster's Collegiate Dictionary*)[1]

The third, most general definition appears closest to its usage in the context of technology audits.

Put simply, an audit is an assessment of the current state of some *system under scrutiny* against *well-defined* criteria. The purpose of this assessment is, as much as reasonable, to quantify the position of the system in relation to a pre-defined target position. This quantification should provide an overall pass/fail grade for the system as a whole, specific ratings for each subsystem, and necessary *corrective actions* to achieve compliance.

Assessments and Audits

It is useful to distinguish among the various types of assessment that we might perform or have performed. At one end of the spectrum is a cursory sweep of the problem space, looking for an overall rating of our existing environment. Where this rating is based upon a comparison of the system with some objective criteria (such as might be supplied by the auditor), this is an assessment of organisational capability or maturity with respect to that subject matter.

Where the comparison is to other companies' performance against similar criteria, this is called *benchmarking*. This measures the maturity of the organisation (with

[1] This dictionary is on-line at <*http://www.m-w.com*>.

respect to the problem space) compared to the industry standard. Both of these types of measurement are useful to management, but do not provide the organisation with sufficient detail to plan the way forward. These are often referred to as *coarse-grained* assessments.

Such an assessment will generally provide an overall system rating, a rating for each subsystem, and perhaps details of one layer beneath that. These assessments are either industry-specific benchmarking assessments (companies like to be able to report where they sit in relation to other companies), or typical of the IT audits provided by large accounting firms. They examine the business aspects of the problem space, such as policies, standards, processes, and responsibilities, but they do not examine the underlying technology or configurations. This is not said to lessen their value, only to clarify their methodology. Such reviews are an important management tool.

On the other hand, a more fine-grained examination and review in the form of a technical audit provides a far more comprehensive evaluation of the problem space. Here we typically see a detailed discussion of the findings, including key recommendations on how to address any compliance shortfall. This type of document is larger and more subjective than a coarse-grained review. Technical audits are more typical of what is seen from small boutique consulting houses with greater technical expertise.

When Is an Audit Not an Audit?

At this point, the line between an audit and a consulting report is starting to blur. An audit should be a passive evaluation exercise. This means two things: it should not make any changes to the system, and it should be independent of auditor biases or vendor alignments. In other words, it should not be recommending specific solutions to a problem—to do so might, at the very least, call into question the independence of the auditor.

The point of delineation is the difference between *observing* and *recommending*. An audit should clearly identify any shortfalls to compliance and any issues that must be addressed in order to achieve compliance. It can and should make recommendations regarding *controls* and *processes* that are required to attain that compliance. It should not, however, make specific recommendations as to how to address those shortfalls in terms of technology, mechanism, or strategy. To do so makes the report a consulting report rather than an audit report.

Where an audit has been requested of a vendor with specific expertise, especially where the sponsor of the report is IT management or where the report is being conducted by internal staff, this line can be more readily crossed.

Technology Audits—The Never-Ending Story

The auditing role is traditionally associated with accounting firms. As is clear from the first two definitions presented above, the word *audit* historically refers to the examination and verification of financial records. This original definition has been extended over time to refer to any detailed study of conformance and compliance to standards, be they financial, legal, or other.

This presents an interesting problem in the context of technology audits: what are the standards against which the system is being assessed?

Auditing financial records is relatively straightforward. Standards of practice for accounting are well defined and have evolved over many decades, even centuries. Information technology, by contrast, is subject to continually changing standards of practice; nowhere is this more evident than at the point of integration: systems administration. Moreover, we each perform systems administration differently; there is no One True Way™. In this context, what constitutes a pass?

Even in the highly restricted scope of a security audit of an IP network, the standards against which that site is to be judged are continually shifting; new CERT/AUSCERT advisories appear almost daily, better tools are developed, and the needs of the user community are continually changing. Any standards that are defined have great trouble keeping pace with the changes in underlying technology. Given this dynamic environment, greater emphasis in the audit process must be paid to the concepts it should embody and the methodology it uses.

A non-technical auditor (someone without the necessary grounding in the subject matter) will find the task of assessing the myriad combinations of technology daunting at best, and overwhelming at worst. These auditors do bring some very important principles and perspective to an audit, however, and people wishing to perform an audit should understand these.

Security Audits

The most common form of audit that system administrators are likely to encounter is the security audit. A whole industry of IT security "experts" has appeared, many of whom provide a service they call an "audit."

Although this is not a text on computer security,[2] it is appropriate to take a short diversion in order to examine this topic as it relates to auditing.

Computer Security

Computer security is far more than a fight to keep The Bad People™ out. It is as much about preventing mistakes and human errors as it is about detecting and preventing intentional abuse.

Computer security is the process of comprehensively managing the overall integrity of the computing environment. As defined in [Ga96]: "A computer is secure if you can depend on it and its software to behave as you expect."

Security, in this context, is generally broken down into three areas:

- **Confidentiality**: Ensuring that the data is accessible only to those authorised to have access.
- **Integrity**: Safeguarding the accuracy and completeness of the data and processing methods.

[2] For an excellent text on computer security as it relates to systems administration, refer to [Ga96].

- **Availability**: Ensuring that authorised users have access to the data and associated assets when required.[3]

In this more global context, computer security includes such things as the integrity of the data backup process, change management, filesystem permissions, inter-application trust, and a whole gamut of issues relating not just to the technology but to the business processes, mechanisms, and controls that relate to systems administration.

Evaluating a site's security doesn't mean running an old copy of SATAN and CRACK and cleaning up the output for management. It is about assessing the underlying practices by which the organisation deals with areas of change. Rather than asking, "Are there any passwords that we could guess easily?" we might instead ask:

> "Do we regularly scan for poor password selection?"
> "Is there a process or mechanism by which easy-to-guess passwords are rejected?"
> "Is there a policy governing selection of passwords?"

Evaluating Security

When evaluating security, auditors refer to the concepts of *controls*, *mechanisms*, and *audit trails*.

A *control* is a point at which the system performs some check over the data or action that is under consideration. It is a choke point through which all actions or data of a particular type must pass, and so provides a guaranteed point at which policy is enforced.

Controls might include asking for a password, checking the location from which a user has accessed the system, or looking up a user's access rights on a database table.

Control points are the points in the system where audit trails are implemented. An *audit trail* is a record of reaching and passing through a particular control. The purpose of an audit trail is, as the name suggests, to provide evidence that policy is being enforced (or breached) at particular points in the system.

A *security mechanism* is the implementation of a security control. This control may be physical, such as a locked door; it may be something codified, such as a password prompt; or it may be something procedural, such as recording an entry in a log book.

The Security Trade-Off

In evaluating security, one is evaluating risk. It is impossible to eliminate risk altogether, but one can always reduce risk or at least control one's exposure to that risk. What is important is the decision as to where to strike the balance between implementing controls and allowing exposures to remain.

Security is inevitably a direct trade-off against convenience. Making something more secure usually involves making it more controlled, which implies more steps in the process. This equates to reduced convenience. (If it is still as convenient for the user, you can bet it isn't as convenient for the administrator!) So, security should be thought of as a point along a security–convenience continuum.

[3] These definitions are from BS-7799 British Standard for Information Security Management.

This is an important point, so here it is again: *security is a trade-off against convenience*. That means that wherever we sacrifice security for convenience we have accepted the *risks* associated with that decision. ("More security" is always possible, but it is always at a cost.) The essence of security management is *risk assessment* and *risk management*.

Risk Management

A corporation must weigh the cost of implementing a particular *security control* versus the risk in not addressing that particular *exposure*. It is important to recognise that every point along the continuum is valid, as long as the decision as to where to reside is an explicit, informed business decision.

A security audit is essentially a risk exposure and risk evaluation exercise. Security audits are not about repairing defects; they are a passive measurement exercise. As with all audits, you are measuring compliance with, variance from, and the adequacy of the company's policies—in this case the security policies. Setting security policy is all about defining where on the security–convenience continuum a corporation wishes to reside. Corporations that do not have a policy are relying on a series of implicit decisions and have failed to evaluate their exposure.

It is not an auditor's role to decide where along the continuum a company should reside, or to set policy. Management must decide the level of exposure and develop relevant policies. The auditor, however, can provide vital information about the organisation's exposures and the costs of addressing those exposures.

Management must make decisions regarding their various exposures—to *accept*, *avoid*, or *assign* each risk:

- **Accept**. Accept the potential cost of exposure against the cost of mitigation.
- **Avoid**. Take steps to avoid or mitigate the risk.
- **Assign**. Take out an insurance policy, subcontract, or use some other mechanism to assign the costs associated with risk management.

These decisions about the various exposures are what define an organisation's security policy.

Beyond Security Audits

Most system administrators are familiar with the concept of a security audit, and no doubt many have been involved in them, or have recommended to management that one should be undertaken at some time. Indeed, the recent explosion of the Internet and its related technologies has brought much attention to tools and techniques for assessing site security. However, audits involve far more than just "running a few tools," and are used for far more than assessing a corporation's security.

Audits come in many shapes and flavours. Wherever a study or review of some system can benefit from the use of the formal concepts and characteristics of an audit (described in the next chapter), wherever you need to measure compliance, and wherever you are wishing to track progress in some quantifiable fashion, an audit is of great potential value as a tool.

Examples of some other audits include:

- **System Performance Audits**. If an application or system is the subject of a Service Level Agreement (and this is increasingly likely with the current trends in outsourcing), then the regular review of system performance against the objectives defined in the SLA is a valuable tool in negotiation and accountability.
- **Quality Assurance Audits**. Increasingly, organisations, including service organisations, are seeking ISO-9000 (or similar) accreditation. This involves implementing a number of procedures and keeping records to track progress through these procedures. Quality-certified organisations must regularly undergo a compliance audit to ensure that the requisite procedures are defined, followed, and maintained. This same concept can be extended well beyond the procedures defined for ISO accreditation.
- **Due Diligence Audits**. Another angle on auditing imposed by management is the external assessment of the state of a part of an organisation or system in order to meet "due diligence" obligations, such as during outsourcing negotiations.
- **System Familiarisation Audits**. If you have recently been given responsibility for a new installation (at least new to you), then your immediate need is to determine the health of, and to gain control over, that environment.

  As a practicing system administrator, you might just want to know whether you're ever going to get a chance to work less than a 12-hour day.

  Gaining familiarity with a site can be done piecemeal, as the need arises. Alternatively, you can choose to begin your new life with a systematic, exhaustive review of the site, save yourself a lot of wasted time, and even impress your new boss with your professionalism and sense of initiative.

  A System Familiarisation Audit is a cursory sweep through all aspects of computer operations, such that you can identify major areas of concern and gain an overall feel for the health of a site.

The Audit of Systems Administration Practices

One of the most insidious hidden costs of systems administration is that involved in staff turnover.

System administrators learn most of what they know through apprenticeship (working with more experienced system administrators), on-the-job experience, and self-education. The result is that each system administrator builds up his or her own personal toolkit containing a combination of scripts and tools they have developed themselves, and those from the Internet community with which they feel most comfortable.

When system administrators start working at sites, they gradually mould the site "into their image," implementing the toolkits and structures that they are most familiar and comfortable with to help them gain control of that site. There is no basis for this, and it is often driven by a lack of understanding of the existing environment and its unique requirements, traits, and history.

I can change accountants without having to spend $30,000 for them to rework my personal or company records to their own standards. Likewise, I can readily change accounting packages. One lawyer can (theoretically) interpret and extend the work of a predecessor. The IEEE has defined a myriad of standards of practice which employers can make use of, and from which their membership must justify any deviation. This same path is one we must move down for systems administration.

As a profession, we must continue to develop standards of practice. In the quickly changing world of information technology, this equates to *world's best practice* (WBP)—a continually shifting standard.

Systems administration is about intricacy and change—the complex interplay of large inter-related components over time. System administrators deal with volumes of highly technical information, hundreds of products, thousands of commands, and the changing individual needs of corporations, departments, users, and customers.

For this reason, we should regularly review the systems administration practices used at a site for their continued applicability and appropriateness to that site. The Audit of Systems Administration Practices (ASAP) is a critical tool in maintaining control over the decay and growing entropy of a site, and the controlled implementation of WBP.

By performing a comprehensive audit of all areas of IT service management, guided by a *body of knowledge* that reflects WBP, we can measure our site against this standard, and against the corporation's business needs. We can then go about addressing the identified shortfalls in a controlled, project-based manner.

The ASAP is, in my opinion, the most powerful tool that a system administrator has for gaining control of a site and being able to proactively improve that site in a way that is measurable and justifiable to management. It is the natural tool for continued maturation of site practices.

When to Audit

System-Driven Audits

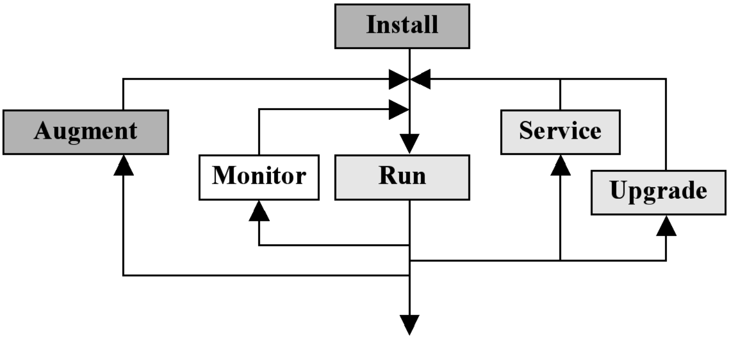


Figure 1: System Life Cycle

Given our definition of an audit as essentially being a test of compliance, it would seem appropriate that such compliance is checked at certain key points in a system's life cycle. So, let's picture a typical system's life cycle, and go from there:

System Installation/Production Acceptance. At the earliest practical point in the system's life cycle we wish to establish a baseline. This is sometimes called the "Day 0" baseline. It represents the system in final production form, but before anyone has begun to make use of it. From such a baseline we can gain much in terms of understanding the basic effect of adding users to that system.

Steady-State Production. Systems undergo a continual process of change, and so it is important to re-audit each major production system on a regular basis against the established baseline. This will greatly assist in capacity planning and compliance measurements against Service Level Objectives. It will also measure the level of decay which has arisen from seemingly insubstantial changes to the system over time.

Service/Upgrade/Augment. After each service or upgrade of the system, we should re-audit to measure the impact of that upgrade. It may be that an upgrade causes an unacceptable performance hit or other resource hit. If this is not caught immediately, we lose the grounds to complain to the vendor. (Of course, the system should undergo regression tests after each upgrade, too, but that's not what this booklet is about.)

Similarly, when the functionality of a host or system is augmented by the introduction of a new product, the system should undergo an audit to ensure that there are no undesired effects caused by the introduction of the new product.

Externally Driven Audits

In addition to the above life cycle–driven audits, there are a number of external drivers for audits, including:

Compliance. In addition to the systems themselves being subject to continual change, the world around these systems is also the subject of continuing change. It is often the case that a system that was built entirely compliant with existing standards becomes noncompliant because of changes to those very standards. Thus, a regular compliance audit is required to assess the level of compliance with current policies and standards.

Familiarisation. In a perfect world, audits would follow the above life cycle, and we would always have a baseline to which we could refer. In the real world, however, auditing usually occurs only after the fact, when it falls within the jurisdiction of a security audit or under some other management or external impetus. Thus, when you enter a site, you will usually not have previous audit data available to which you can refer.

Creating a baseline after the fact is still useful. It serves both to establish that baseline (the result) and to create a detailed familiarity with the system under scrutiny (the process). So performing an audit should be one of the first things you do whenever you enter a new site. It demonstrates a proactive approach to system management, and allows you to evaluate the road ahead.

Emergency. At certain times, it will become apparent that the system has failed its essential purpose in some major way. As our first step, we need a way to assess and quantify the variance and the potential damage. If we have an established baseline, then we can immediately quantify the variation from that baseline, and we will be in a far better position from which to determine appropriate action. Even without this baseline, we must assess the degree of failure and which subsystems are most in need of remedial actions, in order to map out a path forward.

How Often Should Audits Be Performed?

An audit is about assessing compliance. So, the frequency of audits should be directly related to the significance of noncompliance of that system. That is, the more significant a system is, and the more significant noncompliance (failure) of that system against established policies and standards is, then the more frequently that system should be audited.

It's all about risk management. If the potential impact to a business of a particular system failing due to noncompliance is high, then the cost of more frequent auditing is justified by the potential loss. So something like a firewall, which provides a single point of protection for the entire corporation, should be audited more frequently than the security of an internal test server, which resides in a more controlled environment and affects fewer users.

The following values have been suggested as reasonable benchmarks. As always, your mileage may vary.

Firewall security	6 months
Network security	12 months
General host security	12–24 months
Systems administration practices	12–24 months

Who Should Perform the Audit?

The question of who should perform the audit is, and must be, closely tied to the goals of the exercise. If you have just taken over control of a new site (as the senior administrator), you probably wish to determine just what sort of mess it is you are dealing with. In this circumstance (provided you have sufficient experience), you are probably the best person for the job.

You may have a discrete system, such as the backups environment, which you wish to have undergo a formal review. Such a restricted audit should be within the capability of most systems administration teams.

If, however, you are in need of a regular security audit to satisfy management that your firewall and associated security implementation is up to scratch, then using an external consultant (or team) to perform that audit would be more appropriate. Similarly, if you are the IT manager, you may feel that the objectivity an independent consultant brings to the table is of great value.

The ideal situation is for someone external, whom all involved parties trust, to perform the audit. This may be expensive, however, and would require management approval. At this point, you may be losing control of the very audit you recommended in the first place.

There is no single right answer, but don't underestimate the value of a truly independent, objective perspective.

In-House or External Auditors?

Auditors fall into two categories: in-house technical auditors, or external auditors. In this context, a separate audit department should be considered very similar to an external auditor.

There are two classes of auditor, determined by their background. On the one hand are auditors who have come from a technical computing background and made the career transition to auditor. They have in-depth knowledge of the field being audited. The current term for such people is *subject matter experts*.

The other class comprises auditors with an accounting or finance background, and no real technical background in computing. This latter group is often well trained in audit principles, but their background may be financial audits, and they may not have the requisite knowledge of the problem space to perform a meaningful technology audit. On the positive side, these people are generally very eager to listen and learn, are more objective, and can work well in combination with a subject-matter expert to produce meaningful results.

Most important, trained auditors (from whatever background) have an excellent understanding of controls and processes. And in practice, the background of most of the auditors you encounter, whether in-house or external, will combine various elements of these two categories.

In-House Audits. Just because you can't call your audit a "big A" Audit does not lessen its value to the organisation. It all depends on what your reasons and expectations are. If your intention is a technical review of practices (e.g., as a familiarisation exercise or to measure conformance with an SLA), then you may feel comfortable with authorising members of the technical team to perform the audit.

A senior member of the technical team should have knowledge of the problem space (you have been sending them to conferences and workshops, haven't you?) and can generally take an objective stance. This is especially true of discrete subsystem audits, such as an audit of the enterprise backups regime.

If the scale of the audit is large, such as an audit of systems administration practices, then it may be more appropriate to use external resources. Ultimately, independent benchmarking against industry standards can be a very useful exercise, too.

Big N – 1[4] Auditors and Other External Auditors. Many people complain about the quality of audits performed by the "Big N – 1" accounting firms. Experiences regularly recounted are that these firms send in junior robots armed with a questionnaire, an out-of-date copy of a security tool, and no comprehension of the problem space, and

[4] The major accounting firms are often referred to as the "Big N - 1," because they seem to be forever merging.

charge big dollars for an audit which is next to useless. These stories are, unfortunately, far too common to be discounted as the exception.

I encourage all companies that are using external auditors, especially where that audit has been initiated from above, to evaluate the consultants performing the audit, and to satisfy themselves that these people do, indeed, have the requisite understanding of the problem space. If not, raise your concerns with management immediately. There are a number of organisations and associations that specialise in IT auditing standards. Information on some of these can be found in the references at the end of this booklet.

Be careful about complaining to management about the quality of an auditor. This should not be done lightly. It is quite possible that the person has come from a different but technical background. For example, they may be from a mainframe background, and well versed in the underlying technical concepts but lacking in the knowledge of the particular platform being evaluated. If they have a good understanding of the concepts, then you can help them with the keyboard problems, and the resulting audit will probably be of great value.

The Politics of an Audit

No booklet on audits would be complete if we didn't look at the human aspects of this, to varying degrees, intrusive process. Just take a moment to recall your own opinion on auditing before you opened the cover of this booklet. (Hopefully, it has changed for the better over the past 30 minutes.)

Audit Prerequisites

Regardless of who performs the audit, there are a few prerequisites which must be met before you begin:

- **Authority**. Get written permission. Do not commence until you have it![5]
- **Access**. Ensure you have access to everything you need to complete the audit.
- **Attitude**. As an auditor you must put on the hat of an investigator, and separate yourself from your biases.

The Audit Should Be a Positive Experience

Rather than just listing where the system has failed to meet specification, an audit should provide a detailed list of *corrective actions* which, if followed, should bring the system to a satisfactory (passing) grade. This is the difference between an audit being a critical exercise and a constructive exercise.

An Auditor Should Listen and Educate

A major aspect of your work as auditor is education. It is important that all of the parties involved see the audit process as a positive one. Audits can often be viewed by cynical technicians as nothing more than a waste of time destined to uncover what was

[5] There is an oft-quoted case where Randal Schwartz, a system administrator at Intel, was prosecuted for running CRACK without appropriate authorisation.

already known (if only someone had asked you), or, worse, as a blame-allocation exercise.

Be sure to explain and get broad agreement on the goals of the exercise before starting and, once underway, to explain what you are doing and report any significant findings as they are discovered (subject to security clearance, of course). Do not wait until completion, then drop a bombshell at the end of the process in the form of a scathing report. Keep your audience involved throughout the process.

Seek assistance and opinions from everyone—especially those in the trenches. Don't be guilty of the very thing for which you blame other auditors.

Whilst you must be thorough, do not ask your questions in an accusing manner or seek to lay blame, irrespective of any pressure to that end. Keep your audit to noting problems, making recommendations, and suggesting corrective actions.

An Auditee Should Participate

If you are on the receiving end of an audit, it is your responsibility (and is most definitely in your interest) to get involved in the audit. A well-executed audit will have beneficial effects throughout the organisation and so should not be seen merely as something that must be endured.

Your job isn't to try to trip auditors up by hiding a flaw and seeing whether they find it. It is to give them the information they need to do their job well. Tell the auditor where you have encountered problems or a lack of support from management. Tell them where you need more resources. Talk to them. Remember, their job is to listen.

3. Audit Concepts and Principles

The Baseline

The first formal concept we must introduce is the *baseline*. This is the same concept as baselines in project management, source code revision, and other disciplines. A baseline, in the context of an audit, is a snapshot of the system under scrutiny at the time of audit.

The system will likely be undergoing continual change. It is therefore important to perform an audit over as short a period as feasible. When using tools to assist in that assessment, to make the results most meaningful these should all be run in as small a time window as practical.

The primary reason for creating this baseline is, as with all baselines, so that we can monitor and measure changes from that baseline over time. This is an extremely important part of conducting an audit and lies at the very core of a professional auditor's perspective.

Two ramifications come from this understanding:

1. Performing two audits in quick succession should yield (near) identical results.
2. A subsequent audit of a system, when compared to an existing baseline, should reveal all changes (both progress and retrograde steps) since the last audit.

This second point should be explained a little further. If we have quantified our audit, then this reassessment will quantify the progress we have made against the baseline. Management will be able to assess the impact and scale of these improvements against the costs of implementing them. Armed with that knowledge, further improvement programmes can be justified as appropriate.

Evidence

All data gathered during the course of an audit is *evidence*. Just as such evidence is used in legal trials to justify a position or hypothesis of counsel, so too this evidence is used by the auditor to justify a particular assessment of any part of the system.

It is important to retain copies of all evidence used in reaching your assessment. This evidence may include printouts, interview notes, system documentation, copies of key files, and system information gathered both manually and via software tools.

Evidence, in the context of auditing, has another purpose. It forms an integral part of the baseline against which we seek to judge progress, and hence will be used in future audits. Evidence gathered during an audit may be highly sensitive (especially that gathered during a security audit), and so must always be maintained with appropriate security controls.

It makes good sense, therefore, to store all of this information in an isolated environment during the audit and onto secured archival media once the audit is complete (removing any on-line copies). A laptop computer is useful for gathering data during an audit, and a CD-ROM burner is valuable for archival storage afterwards.

Whatever written notes you take during interviews or system inspections (or, for that matter, in the train on the way to the site) also constitute evidence, and should be similarly secured and kept on file at the completion of the audit. All such notes should be dated. This cannot be emphasised strongly enough. These notes form a critical part of the evidence of an audit, and in directing the flow of the audit investigation. Dates help.

Some Audit Principles

There are some key characteristics of audits, which should be understood right from the outset:

- **Quantification of Assessment.** It is important to attempt to quantify (reduce to a measure) as many aspects of the audit as reasonable. I use the word *reasonable* here because it is more important to get even a qualitative assessment down on paper than no assessment at all. We will come back to this issue in detail later in the booklet. Where practical, each major subsystem under review should be rated separately. This, in combination with the defined importance of that subsystem, should directly lead to management commitment to improvement programmes.
- **Consistency of Assessment Criteria and Rating System.** It is important that, as much as is reasonable, the audit be related to some independent, well-defined set of assessment criteria, such that two different (appropriately skilled) people could audit the site against the same criteria and obtain a similar (nearly identical?) result. Again, the word *reasonable* is used here because as an audit delves deeper into technology it will inevitably be influenced by the particular experience and views of the auditor.
- **Independence and Impartiality.** It is important that the assessment be as objective as can be achieved under the circumstances of the audit. The more objective the auditor, the more valuable the result. Where there is, or may be, a bias held by the people performing the audit (this may be unavoidable), it should be clearly stated in the audit report.
- **Transparency.** In order to meet the goals of independence and impartiality, it is also necessary for the audit methodology (including any checklists, tools and evidence) to be open to scrutiny and review. Each audit should provide a statement of the audit's methodology and access to copies of the assessment criteria used.

- **Verification.** It is important to keep all supporting evidence gathered during the course of the audit so that the audit itself is open to verification.
- **Completeness.** The audit report must clearly define the scope of the audit. Having defined that scope, it must be the reasonable expectation of the reader that the problem space within scope has been thoroughly investigated. Any exceptions to this must be clearly defined in the audit report.
- **Security.** Maintain thoughtful and appropriate security over any evidence you gather, and any interim or final audit reports. Reports should only be distributed to authorised recipients, and evidence must be suitably secured against prying eyes.

Quantification

Quantification is a complex area, and so deserves a little more discussion.

Continually changing technology and an ever maturing understanding of the subject matter alter our view of what is a desirable state and, hence, our assessment criteria. *Best practice* refers to keeping up with a continually evolving notion of how we should be operating a particular business area, or *system.*

Thus, your checklists, questionnaires, and software tools, as well as the grading system used, should be under continual review. This means that a system may pass one audit, but then fail the following audit due to changes in the audit criteria. This is not a bad thing.

Having created a baseline, it is important to *convert* the results of the previous audit (i.e., subject the original audit evidence to the new criteria and standards of practice). This way we are comparing apples to apples, and so may identify more clearly what progress has been achieved.

4. The Context of an Audit

Assessment and Repair

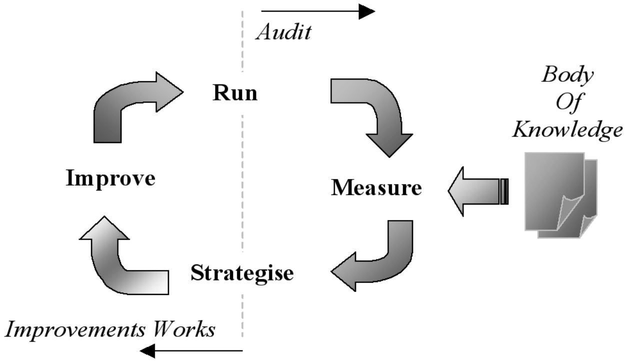


Figure 2: The Continuous Improvement Life Cycle

The steps of the continuous improvement life cycle are best broken down into two groups: system inspection and assessment, and the subsequent improvement works, or simply *assessment* and *repair*.

An audit forms one half of a *total quality management* life cycle. It is the process by which we measure how we are doing, and what progress we have made against objectives. The other side of this coin is the *controlled improvement* process, where we effect changes to our practices in order to address a deficiency, or improve a metric.

The four steps of a comprehensive system inspection and assessment are:

1. **F**amiliarisation and agreement.
2. **E**xamination (conducting interviews and system inspections).
3. **A**ssessment (reviewing evidence).
4. **R**eporting (reporting findings and recommending corrective actions).[6]

The result of this process is a comprehensive audit report that presents a detailed review, ordered by topic, of the system under scrutiny. It provides an assessment of where the system presently is, where it should be, and a series of recommended corrective actions for reducing any gap between the two.

This audit report should then form the basis for one or more *controlled improvement programmes*. These are discussed in a later chapter.

[6] I like the acronym that this process creates—it is in line with people's perceptions of an audit, and so is easily remembered.

The primary attribute of the *assessment* phase is that of *completeness.* Here we seek to walk through the problem space methodically, probing every cavity, gathering evidence, and gaining a detailed understanding of each component or aspect of the system.

By contrast, the primary attribute of the *repair* phase is *predictability.* Having gathered a list of corrective actions (CAs) which address the gap (chasm?) between where we are and where we should be, the emphasis must now be on implementing these CAs with minimal impact on the user community. The systems we are studying and repairing are usually production, if not mission-critical, environments where downtime costs real money, so such planning is essential.

Our planning can be thought of as converting a *topical* analysis into a *temporal* project plan.

The Audit Process

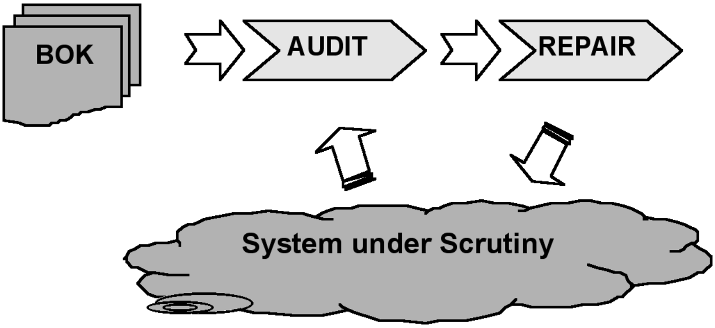


Figure 3: The Context of an Audit

As you will come to see in this booklet, there are two separate aspects to performing successful, meaningful audits. The first is an understanding of the concepts, principles, and procedures for performing an audit (the *audit process*). The second is the development, familiarisation with, and use of a *body of knowledge*—the standard that guides your examination and assessment of the problem space.

The Body of Knowledge

In order to perform a meaningful, thorough analysis and assessment of a particular problem space, we must first define that problem space.

The key attribute of our assessment must be *completeness.* In order to meet this objective, we must be sure we have performed a methodical, structured sweep of the entire problem space. Vital to our success, then, is that we have previously created (or obtained) a comprehensive breakdown of the problem space—the *body of knowledge* (a.k.a. *topical assessment checklist*).

The *body of knowledge* (BOK) is a descriptive text, highly structured in its organisation. It should be structured as a topical breakdown of the problem space, with a list of imperatives, a statement of World's Best Practice or the desired state, and a definition of what constitutes acceptable standards for that topic.

Such a breakdown should be conceptual, and not one that embeds the underlying technology—the latter changes over time, and so would create a need to convert historical checklists in order to re-evaluate against an established baseline.

It is only with experience of the problem space and the sharing of these knowledge bases that we can begin to arrive at industry-wide standards for such assessments.

There are many ways in which we could look at the problem space of systems management and related fields. Several such topical assessment checklists are listed in Appendix A. You should find these a useful starting point in drawing up your own such lists.

Controlled Improvement Programmes

The audit is an assessment of compliance. Its output, therefore, is a list of instances of noncompliance. Such an assessment is of limited value to management if it does not provide a quantified method for addressing these compliance shortfalls. The more meaningful audits make this process a positive, constructive one by listing *Corrective Actions* rather than only listing items of noncompliance. So, the output of an audit is a set of CAs, ordered by *topic*, and possibly also by *priority*. This information is very important, but not sufficient.

To bring a system into compliance, by implementing the CAs listed in the audit report, requires a number of decisions from management, and usually requires effecting change on a non-trivial scale to a production environment.

One of the most difficult and complex tasks is to effect repairs or improvements to a running production environment. If it is a mission-critical environment, then this is all the more complex. People not charged with the management and support of such an environment do not understand why this is so. The audit report is the first step in this process of shepherding change.

One outcome of an audit is a set of corrective actions which must be performed to bring the system into compliance with the standards against which it was measured. (No actions are performed during the audit—it is merely a passive measurement exercise.) Moreover, the resulting list makes no measurement or suggestion as to the complexity or cost of performing these actions; this is left to the technicians who manage the system.

It is important, therefore, that a technician (such as a system administrator) draw up a project plan that implements the recommendations (or at least those approved by management) with minimal disruption to the production environment. It is often best to group actions into smaller, more controllable technical improvement projects. This planning thus results in one or more controlled improvement programmes.

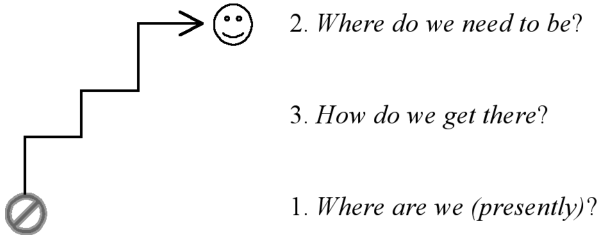
We will look at these programmes in detail in a later chapter.

5. The Audit Process

An audit is an assessment of some *system under scrutiny*, ***with the intention of improving that system in those areas that are identified by the audit as requiring attention. So, we can think of an audit as the process by which we seek the answers to three closely related questions:***

1. Where are we?
2. Where do we need to be?
3. How do we get there?

These three questions of the audit can be visually represented thus:



2. *Where do we need to be*?
3. *How do we get there*?
1. *Where are we (presently)*?

Figure 4: The Audit as a Journey

The first question seeks only to assess (and identify shortfalls with) the present situation. Without having answered the second question, there is no basis for that assessment. The third question is what makes the audit a constructive exercise.

The Audit Time Line

Conducting an audit, like so many aspects of computing, is based upon a "peeling the onion" model.

An audit is an investigative exercise. Each audit is unique. After all, each system is unique. An audit is the process of taking a fixed body of knowledge and comparing it to a unique system. This is not a black-and-white activity.

Thus, the general process presented here should be taken only as a guide. As you progress into more detail, you will often discover evidence that requires returning to an earlier phase or step (as described here). This is the normal process of feedback loops and stepwise process refinement.

Do not think of the phases described as distinct steps, one following on from the other, but as composite activities that proceed in parallel. The diagram below illustrates the degree of overlap between the phases of an audit:

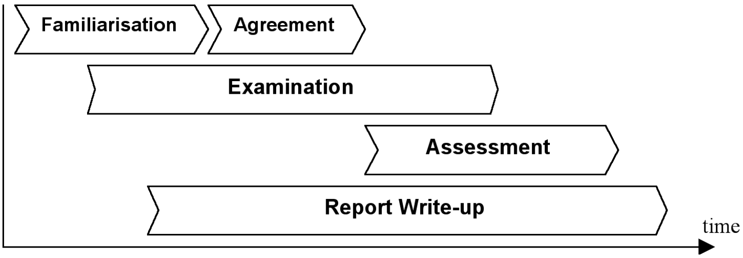


Figure 5: Approximate Audit Time Line

Distribution of Effort

What is perhaps most surprising to someone new to audits is just how little actual time is spent in examination, in proportion to the other activities that make up an audit. The above diagram illustrates the overlapping of phases across the course of an audit (in time). The table below examines the effort required by various audit activities as a proportion of overall effort:[7]

Component	*% of Time*
Preparation	10
Reviewing policy/docs	10
Talking/interviewing	10
Technical investigation	15
Reviewing data	20
Writing report	20
Presenting report	5
Post-audit actions (clarifications)	10

These figures provide much of interest. For example, you might be able to better estimate the time required to perform an audit, based upon the size and complexity of the system under scrutiny. For a security audit or audit of systems administration practices, you might determine (guess) that it will take 4 hours per major device and 2 hours per minor device. A network of 15 major hosts and 5 minor hosts could then be estimated at 70 hours. But this only accounts for 35 percent of the total audit effort.[8] Thus, the audit would likely take 200 hours or more.

As with any such complex activity, estimates improve over time, based upon experience.

[7] This table comes from Dan Farmer and Wietse Venema's workshop on security auditing [Fa96].

[8] This figure is derived by combining the steps of technical investigation and reviewing data.

6. How to Perform an Audit

So, the context has been set, the parameters and process discussed. It is time to examine the steps and activities of an audit in more detail.

Remember, what are presented here are guidelines, not rigid rules to be followed to the letter.

Step 1: Familiarisation

Systems administration is about dealing with complexity. Five sites can have identical hardware and software, and yet each will be unique. There is no One True Way™ for managing a site. Each site's system management practices are the result and culmination of the history of that site. Just because a site is not being managed the way you would like it to be, that does not make the existing management practices wrong. By the same token, the continual stream of gradual changes over time that typify systems administration is also the key contributor to the level of entropy at the site, and hence an indication that a systemic review of practices is due.

Before we can accurately assess a site, it is vital that we learn the nature, context, and history of that site. Only then can we come to an impartial conclusion as to the correctness of the management practices employed, given their present environment.

Further, before we can begin the audit proper, we need to come to an agreement with the *audit sponsor* as to the objectives and priorities of this exercise. We cannot do that without some knowledge of what we're dealing with.

So, the first step is to gain a cursory understanding of the problem space—its size, shape, complexity, age, and any likely indicators of entropy.

In order to obtain the necessary context for the audit, and a feel for the site, an initial interview should be organised with key personnel, including:

- Someone from management (the IT manager or the manager's delegate). This person should be aware of management's expectations from the audit, and management's perspective on their IT environment.
- Technical staff who understand the architecture and major functional layout of the environment. An important part of this initial interview is to gain a solid understanding of the entire IT environment, and the scope (in terms of hosts, network devices, applications, etc.) that is to be considered by this audit. Technical staff can help provide this understanding.
- Someone with a number of years' experience of the site. The nominal "site historian" is a vital link to the reasons that something was done in the past. The most difficult question to answer, and the most valuable one to

attempt to answer, is "Why?" The historian is your vital link to the gradual nature of change within the environment under scrutiny.

- Users of the system. Always obtain the perspective of key system users, from each of the major user communities.

This initial interview should peel the first few layers of the onion, to create a basic context for the audit. It may be more appropriate due to the variety of topics being discussed, or people's availability, to conduct a number of interviews, rather than just holding one large meeting.

It will often also be of great value to review any printed documentation that exists, perhaps prior to the first interviews. The interviews will also uncover further documentation which can then be obtained for review.

The exact nature of the system will vary greatly, but as an example, here is a suggested list of topics to gain a general context of the system:

- Company overview
  - Business
  - Products
  - History
  - Organisational structure
- IT organisation
  - IT overview
  - Network topology and overview
  - Computing platforms
  - Business applications
  - Availability requirements
  - Major hosts and network devices

Step 2: Agreement

The intention of this step is to establish an agreement with the *audit sponsor* which answers the question, "What is this audit meant to achieve?" (It's always nice to establish the conditions for success before we begin an assignment.)

What are management's expectations from this audit? What are management's expectations for the site? How much effort will be involved in conducting this audit? How much of an impact will this audit have on people's time? What other resources are required to conduct the audit?

It is important to have obtained a basic level of familiarisation with the system in question (the previous step), so that we can now proceed to establish meaningful objectives, and hence this agreement. (We can't reach a reasonable agreement without that prior familiarisation.)

The agreement step is generally conducted as a meeting with the audit sponsor, having reviewed all pertinent information obtained during familiarisation. It is helpful at this point to explain the basic methodology to be used, milestones, time frames, etc., and obtain broad agreement and management support of these objectives right from the outset.

This agreement should be a written one. This allows both parties, the auditor and the audit sponsor, to ensure that expectations have been set correctly.

Step 3: Inspection and Evaluation (Conducting the Audit Proper)

Finally, we get to the real meat—the actual system evaluation.

Our goal in performing this evaluation is to take a snapshot of the system, so that we can compare this snapshot to the last one taken (the baseline) or, where one has not yet been created, to create that baseline.

The key attribute of this phase is *completeness*. This cannot be emphasised strongly enough.

In order for an audit to be of genuine use, the involved parties must have confidence that all relevant data was gathered and considered and that the conclusions reached were reasonable and correct. You must keep in mind at all times that the audit report will be read by a number of people of varying levels of technical ability. Thus, you will have to justify to each of these readers any conclusions drawn, or your recommendations will have a hard time reaching implementation. We will return to this point in detail when examining the audit report, but I bring it up now because you must be sure to gather and collate all relevant evidence, and reference this evidence in your findings.

There are three basic techniques or tactics for investigation and collection of evidence: interviews, manual system inspections, and automated system probes. These techniques should be supported in the form of interview questionnaires, inspection checklists, and probe software. These support tools should be tailored to the system as appropriate.

Pre-Audit

Before the audit can commence, you must prepare. It is important to refresh your memory with the methodology, checklists, interview questions, and tools that you will use. Re-read a previous audit report you have written or a previous audit report for the system you are auditing. Get yourself into the mindset of an audit. This will be an intensive couple of weeks.

Ensure that you have the latest version of your questionnaires and checklists. Each time you perform an audit you will gain more knowledge, which should be incorporated into these documents. You may alter the structure of your inspection to reflect new knowledge of the problem space. All this means that you must review your checklists and questionnaires to see that they are complete and ready for use.

Prepare your tools. We have not yet discussed the role that software tools play in system inspections. This is covered in a later chapter. It is important to familiarise yourself with the tools you intend to use, and to ensure they are up-to-date. Security audits, for example, will often make use of Internet security tools, such as automated system probes. It is important that these reflect the latest security flaws for which advisories have been issued. Not to do so diminishes the value of the audit.

Be sure that you can trust your tools. Even more important than obtaining the most recent releases of tools is to be able to put your full trust in the authenticity of the tools that you do use. This means protecting the integrity of those tools and any systems you run them on.[9]

Round One Interviews

Having already conducted a first interview, the familiarisation interview, you should now have a reasonable understanding of the system in general and the (intended) relationships within the subsystems. You can now tailor your generic questionnaire to the environment at hand, and begin to gather more detailed information about the organisation, history, people, duties, policies, and strategies. Whilst the systems themselves will often provide clues to the skilled investigator, this is information that can only be extracted reliably by interviews.

It is also worthwhile to use an interview to ask some questions which you can find (or have found) the answers to by system inspection. This serves to verify the gap between system behaviour and people's perceptions of that behaviour.

Round One System Inspections

This is the first pass at the systems themselves. Using a combination of tools and manual system inspections, you must walk through the system, topic by topic, gathering evidence. This is an investigation. You must not rely on the accuracy of anything you are told. Verify all assumptions.

Remember that "the system" includes documentation and other non-electronic components.

Don't be afraid to wander "off topic" during your inspection. Follow leads, make notes of things that need to be checked or verified. When you have exhausted the path, return to the topical checklist.

This step is where you can make use of software tools to automate the data collection process. This improves the consistency of the results and reduces the time window required for data collection. This time compression may be important, especially if you're gathering performance data. We will look in detail at system inspections and software tools in later chapters.

Step 4: Preliminary Assessment

Having interviewed all key people and gathered volumes of raw data from the various components of the system in question, it is time to organise and review that evi-

[9] Maintaining your audit tools on CD-ROM is a cheap way to achieve that goal. CD-ROM burners and associated media are sufficiently low in cost to warrant their use the moment you proceed beyond a single audit. A CD-ROM may also prove to be the most practical archive mechanism for the evidence you gather from an audit.

dence, and to come up with an assessment of the site. The assessment should be in terms of the three questions we posed at the beginning of the audit process: Where are we? Where do we need to be? How do we get there?

We present this information in three parts:

- The present situation.
- Problems with the present situation.
- Recommended corrective actions.

This assessment should be organised according to the structured walk through the problem space that we have already performed based upon the topical assessment checklist.

This analysis is presented using the principle of progressive disclosure. The assessment process itself tends to be more ad-hoc, though, and is a combination of top-down and bottom-up examination of the data. Do not expect this process to be a smooth, orderly walk through the evidence. It should start out that way, but expect to be distracted as you uncover issues. It is always best to document these issues whilst they are fresh, then return to where you were.

Remember also that there are many ways to achieve the same (or similar) goals. Keep an open mind as you examine the system. Just because they didn't do things the way you like doesn't make their way wrong.

Iterate

The evaluation itself will take at least two iterations: the first round of interviews and system inspections and subsequent analysis will no doubt raise many questions that must be answered. To resolve these, we will need to interview and inspect further, and to re-evaluate previous facts in light of this new knowledge. This process may go through several iterations.

A general hint: use two pads during the audit process—one for writing down your notes, and one for recording issues that must be checked later. You can then check off each of these action points at a later time, ensuring that nothing slips through the cracks.

Step 5: Reporting

Having gathered the relevant evidence, you'll need to collate and analyse that information and present it in the form that is most likely to see any recommendations given their appropriate weight. That form is the audit report.

An audit report is a structured document which provides progressively more information to the reader. A manager will only read as far as the first section, and will expect his or her delegates to review the recommendations carefully. A technician will read the recommendations with great interest, but may not agree with all of them. Be prepared to discuss this in detail.

We will examine the audit report in greater detail shortly.

So, What Are We Looking For?

This question is at the core of what we do when we are auditing a system. What exactly is it that we are looking for when we examine a system? How do we determine whether the system has met the criteria defined for that particular subsystem? How do we define the baseline performance metrics for that subsystem?

The following list represents some of the perspectives from which you should be examining and evaluating each topic within the problem space:

- **Policy.** Are appropriate policies defined and promulgated regarding this topic? Are they pertinent and up-to-date?
- **Standards.** Are appropriate standards defined and adhered to? Evaluate both external standards (e.g., financial, legal, fiduciary, ISO-9000) and internally defined standards (e.g., Standard Operating Environment).
- **Processes.** Are business processes and procedures defined? Are they appropriate? Are they effective? AHave they been allocated adequate resources? Are they used consistently?
- **Responsibilities.** Have people been officially allocated the responsibility for implementation and oversight of policy, standards, and processes? Do they have the authority to carry out their assigned functions? Are they accountable for their assigned responsibilities?
- **Controls.** Are there appropriate controls, mechanisms, and audit trails in place to confirm compliance to standards, processes, and policy?
- **Results.** Is the organisation achieving the desired results with consistency?

7. Interviews

The interview is generally seeking answers to two things: (1) the question "Why?"; and (2) questions of organisation, history, policy and the like, which relate to the people side of the equation rather than to the technology side. Do not underestimate the importance of these issues. *Organisational issues will prevent technical solutions.*

The Familiarisation Interview

You should hold the initial interview prior to the audit commencing. Its purpose is to provide some basic context and understanding of the environment (problem space) that will be examined.

This is achieved by gathering key members of staff, both from within and external to the IT organisation, in order to gain a well-rounded perspective on the size and shape of the problem space.

This interview has been described in detail in the previous chapter.

The First-Round Interviews

The audit activities begin in earnest with a series of first-round interviews. These should be one-on-one interviews with the people identified by the familiarisation interview and audit preparation as significant to the problem space. The intent is to solicit their knowledge and opinions with respect to all aspects of the problem space for which they can speak with knowledge or experience. By making these interviews one-on-one, we provide an open forum for exchange, away from the external influence that might be felt by a subordinate should their manager be present when they are being interviewed.

In general, a first-round interview should comprise two sections: a standard set of questions, and a more interactive interview section. The first section can be performed either via a written questionnaire or in person. I prefer the in-person approach, as it provides more room for the interviewee to expound and diverge from the basic question, providing more useful information than might be obtained from a written questionnaire.

The purpose of the standard set of questions is to obtain an idea of the degree of consensus of understanding that exists amongst staff and key users as to the system in question. By asking a common set of questions, we can gauge the reality of the organisation rather than the paper world drawn by policy and procedure documents. It also provides a defined time for responses—not wasting your or the interviewee's time.

When asking the standard question set, be careful not to colour the questions with the knowledge and feedback you have acquired from previous answers to these questions. Asking exactly the same question is a good way to: (1) verify that a number of people agree on the present situation, and (2) give each of them the same objective stance from which to offer variations.

This standard questionnaire should be created/customised for the site during the pre-audit phase, once you have been through the site familiarisation interview and have a basic feel for the problem space.

Once you have completed the standard questions, the interview continues with free-form questions, based upon the answers you received during the interview, previous people's answers, your knowledge of the environment, or any hypotheses you are already forming about the environment.

It is important to practice active listening. Repeat back to the interviewees, in your own words, what they have said. This gives them the opportunity to clarify their answer for you.

Subsequent Interviews

Follow-up interviews are primarily about clarification of information you have gathered from previous interviews and system inspections. During a system inspection, you will often encounter things that don't appear to make any sense at all. These are generally related to the history of the site, and are indicative of entropy. Asking people about them may be the only way to gain an appreciation of their continued significance.

Interview Techniques

Where possible, keep interviews one-on-one. This has two advantages: (1) you reduce the impact of the audit on the organisation, and (2) you provide people with an environment that is most likely to elicit open and honest exchange of information.

A counterpoint to this approach is that interviewing a team together provides an environment where one comment might trigger another comment from someone else, uncovering more information than might have otherwise been discovered. Knowing when to use which technique is a skill that comes with experience.

It is important to listen carefully during an interview. This can cause a conflict with the goal of capturing the maximum amount of information. If you take sparse notes, you can pay more attention and make more eye contact with the interviewee, but you may miss some information that is important. You can introduce someone else as a minute taker or record the session—but be sure the interviewee is comfortable with such an option before using it.

Your role is to be the guide through the interview process. Give the interviewee ample room to wander off-topic. Make them feel comfortable and listened to.

Who to Interview?

Your initial interview, the site familiarisation interview, should have a diversity of people present. This interview was discussed in the previous chapter. Subsequent interviews, generally with only one or two other people present, should include the system maintainers, management, and key representatives from the user community. This should give you a good enough cross-section to provide an accurate perspective on the organisation.

When performing a review of a major technical system, be sure to interview all support staff, especially the people on the front line. Failing to include these people can lead to alienation, which in turn can create resistance to both the audit process and the implementation of any results. Worse, failure to talk to the front line will generally lead to information gaps and result in an incomplete or incorrect audit.

8. System Inspections

System inspections form a major part of the audit process. Inspecting a system is the surest way of identifying weaknesses and gauging the level of decay and noncompliance.

As with all other aspects of the audit process, the system inspection is performed by drilling down into progressively more detail. Start with a simple cursory sweep of each system, building up information on the key components of that system and their interrelationship. What is the trust between subsystems? Is the interface between them well defined? What is the nature of the relationship?

Active Versus Passive

There are two major approaches to system inspection. There are *passive inspections*, where you look carefully around the system, avoiding anything that may affect availability; and there are *active inspections*, where you attempt to prove the existence of security holes and other problems through exploitation. The latter can be very useful, but be sure to warn people and get management to sign off before you do *anything* that might affect availability or otherwise breach policy. Both types of inspection can benefit from automation to improve consistency and to reduce the time window required.

Passive Inspections

Passive inspections involve a manual investigation of the system, at all times staying "within the lines." In this mode you are a visitor, and you must not do anything that may breach corporate policy or potentially affect system availability. Most of your system inspection time should be spent in passive mode. This is where you examine data in detail and follow your nose, developing and proving hypotheses regarding compliance (but not proving them in such a way as to affect availability).

The system inspection can lead you down a number of paths. It is important to follow these paths, but it is also important to maintain focus by using the body of knowledge (BOK) to guide your investigation. Make notes on points that require follow-up, so that you can return to them later.

For example, on UNIX, it is prudent to examine root's `crontab` in detail. This may then require examining each of the scripts invoked by `cron`, to examine the security implications of these jobs being run by root. In order to do this, you will need to be proficient at shell and at reading other people's spaghetti code.

Active Inspections

Active inspections, by contrast, involve proving hypotheses by trial. The advantage of this approach is that you have irrefutable proof as to the existence of the problem. (Managers tend to listen when you show them their cracked password!)

The disadvantage is that you can directly affect data availability (denial of service, service degradation), data privacy (such as running CRACK to prove that passwords are not secure), and data integrity. It is vital, therefore, that you inform the audit sponsor well ahead of performing any active inspection and that you get their written approval before proceeding.

This gives sponsors the opportunity to warn people if they so choose (they may feel that warning is inappropriate, depending upon the nature of the probe),[10] schedule the event for a convenient time, or simply say no to one or more of the intended probes.

Of course, you should always ensure that adequate backups and other recovery procedures are executed prior to commencing any active inspection.

The recommended method for dealing with active inspections is to perform as much of it as you can in passive mode. Whilst studying the system in passive mode, draw up a list of the active probes that you believe need to be performed. Present this explicit list of probes, along with any explanation of likely and potential effects, to the audit sponsor. Remember, proceed only after you have gotten their written sign-off.

The distinction between passive and active may not always be obvious, but you must keep it uppermost in your mind. In practice, there should be little need for active inspections. Most of what you can learn from an active inspection would be readily visible to a trained auditor using a passive inspection. On the other hand, a demonstration can be a valuable technique in getting a message across.

Automated Probes

Given that one of the major goals of an audit is the creation of a baseline, with subsequent measurement of progress (variance from that baseline), it seems obvious that the more we can automate the collection, collation, and analysis of the evidence maintained in the baseline, the more consistent our data set and the more accurate our audit results will be.

Where we are measuring application or subsystem performance or some similar transient metric, the automation of this measurement is critical to the efficient, timely, regular collection necessary for meaningful results. Automation also directly reduces the effort required by the auditor, and so reduces the cost of auditing. This may, in turn, lead to more frequent audits, yielding greater control over the improvement process.

In addition to the use of tools for data collection is a more sophisticated use—that of data analysis. As our familiarity with the problem space improves and we better understand what constitutes acceptable practice, then automated tools to ensure compliance can be employed to greatly reduce the auditor's manual workload.

Tools such as Tripwire and HCB perform extensive data collection and baseline management, including notification of changes from the baseline, but do not perform any analysis on the data collected. By contrast, tools such as COPS, with their in-built understanding of what constitutes acceptable practice, not only gather information but also analyse that information for relevant instances of noncompliance.

A final word on automation: *you can't automate what you don't understand*. Until you

[10] Refraining from informing staff of an impending active inspection is one way to verify that your detection systems are working.

have a manual procedure for performing the data collection, collation, or analysis activity, it is meaningless to attempt to automate that procedure.

Data Storage and Security

You may be collecting large volumes of data during this process, and it is important to gather it into a secure repository for later analysis.

It is quite likely that you will be given administrator-level privileges during the course of the audit, so that you will be able to gather the required information. This is not unreasonable under the controlled circumstances of an audit. However, it is bad practice, and also potentially a large amount of effort, to extend that level of trust to some unknown (to them) data collection machine that you bring with you. Rather, you will generally need to gather data locally, then transfer those data collections across to your *secured audit host* via an appropriate means, and maintain the security of that host at all times.

9. The Audit Report

The audit report is the culmination and most significant aspect of the audit process. It is the primary communication tool for transmitting your results and findings to the wider community. If you cannot communicate those findings and needs effectively, then you might as well not have performed the audit.

The audit report is a comprehensive, highly structured document. It must present the audit results in a succinct, clear, meaningful communication. It will be read by upper management and by technically proficient staff. It must be literate and highly readable. It should serve, not just as a critique of the system, but also as an educational document.

Know Your Audience

When writing a technical document, it is vital always to keep your intended readership in mind. Telling upper management about the need to upgrade to release 17.8.8 of supermail is a futile, overly detailed gesture. (The release number will have incremented by the time they read the report, anyway.) Of course, noting that the version being run is six years out of date with respect to security holes may be relevant to effecting change.

An audit report presents a problem in this respect; it is read by a variety of people, at differing levels of technical competence and with different agendas and goals.

The basic rule for dealing with a multi-scoped document such as this is the rule of *progressive disclosure.*

Progressive disclosure is the technical way of saying "Peel the onion." Your report begins with a cursory coverage of the audit results, just highlighting the current state (pass/fail), how much effort will be required to repair it, and how best to proceed—a typical executive summary.

Each subsequent section of the report then provides progressively more detail of the audit results. Even a technical reader of the report will gain valuable context from this telescoping approach, whilst a less technical reader will just stop reading when they have had enough, and may perhaps refer to later sections as required.

The audit report, in its more detailed sections, should reflect the topical breakdown of the audit scope.

Where the resulting report is large, it may be appropriate to split it into several documents, each with a different scope or audience. Thus the executive summary may be presented by itself, and the detailed findings presented in a second document.

A Walk Through an Audit Report

A completed audit report should run between 10 and 100 pages, depending upon the complexity of the system under scrutiny. The basic structure of the document consists of some standard sections up front, the main body of the report, and a few closing appendices.

A cursory run-through of the structure of a typical audit report would reveal:

- Title Page
- Table of Contents
- Executive Summary
- Audit Goals and Objectives
- Audit Methodology
- Audit Context: Organisational and Technical Overview
- Main Body, Including Corrective Actions
- Appendices
- Glossary

The Executive Summary

The executive summary is a succinct synopsis of the findings of the audit, as detailed in the later sections of the report. It is to be read by upper management, and must draw their attention to the essential messages of the audit.

Because of its significance, and the intended audience, it is perhaps the most difficult part to write. It can be no more than one or two pages in length. Any more and it may not be read to completion.

It should consist of:

- A statement of the scope of the audit. What is the *system under scrutiny*? It might also refer to major systems or components which were declared beyond the scope of the audit.
- A statement of the compliance metric (the final rating). This may be broken down to include the rating of each major subsystem.
- A list of the key findings and recommendations. Keep this list short. A maximum of six recommendations is a good rule of thumb.
- An impact statement. The potential negative impact (in management terms) of the system continuing without addressing the defects found, and the potential positive impact (e.g., savings, risk reduction) and the costs associated with implementing the recommendations.

Expect to make several rewrites of this section. If possible, you should have someone review it (with the appropriate security clearance, of course). Given the target audience (upper management), it is important to ensure that you have not accidentally used jargon or assumed an inappropriate level of knowledge on the part of the reader. For this reason, you should also get a non-technical person to review the executive summary.

Audit Goals and Objectives

In this section, we clearly state the agreed objectives of the audit exercise. You will remember that after our initial familiarisation we held a meeting with the audit sponsor to establish these objectives.

From that meeting we should have arrived at an agreement/definition of:

- The objectives for the audit. These should be listed in priority order. Why was this audit initiated? What is the expected outcome? Is there a particular business driver for the audit (e.g., SLA compliance)?
- A definition of the *system under scrutiny* (a.k.a. the *scope* of the audit).

  What component subsystems/hosts/networks were considered part of this audit? What are the boundaries of this audit? Diagrams can often be a useful tool for setting this context.
- An explicit statement of any components which are beyond the scope.

This list forms an important context for the readers of the report.

Audit Methodology

This should be a statement of the methodology used and any variations from it. This section should contain:

- A reference to the checklist/body of knowledge used (and its version number).
- A list of the tools used (with their version).
- The people interviewed.
- The elapsed time (and dates of data collection).
- Any statement of bias or other auditor statement.

The auditor statement is an important inclusion in the audit report preamble. If the auditor has prior experience with the site or its staff, then these will colour the auditor's views to some extent. It is, therefore, vital to the integrity of the audit report that any such influences are stated.

I have used a variation on this theme of disclosure: to counter people's natural perception of the audit as a negative exercise, I tried to set a positive tone in the very first paragraph of the report! I opened one executive summary with the following:

> I must begin this report with a general observation. The staff in XXX are extremely capable people whom I have a great deal of professional respect for. The job they have done and are doing is exceptional. Their ideas and implementation regarding the management of a distributed UNIX network is first-class work.
>
> The nature of a security audit is such that it is a generally negative document, focusing on what is yet to be done, and what must be accomplished as a matter of priority. It leaves little time to reflect on the positive aspects

> of existing system management practices. I therefore feel that part of my duty is to take a few moments at the beginning to make it clear to management that they are extremely fortunate to have a team as skilled and dedicated as the XXX system and network management team. It is clear both from my observations and from my discussions with them that the only reason that much of what is presented here has not been already implemented is purely a lack of resources.

Now, although it may not be obvious, this is actually a quite blatant statement of bias! It also has a very positive effect on the reception of the entire report, thus increasing the likelihood that its recommendations will be implemented. Setting a positive tone in the first couple of paragraphs is a very powerful tool for improving the receptiveness of the reader.

In order not to detract from the flow of the report, it is often best to move the raw details of checklists, tools, and versions to an appendix, rather than include it in-line in this section.

Audit Context—Overview of Organisation and System

This section is the first layer of the onion. It is a cursory sweep across the entire problem space, identifying the component subsystems, their significance (weighting), and general findings. It sets the scene for the main body, which follows directly.

The main purposes of this section are to act as an introduction for someone who is not familiar with the problem space in detail (e.g., management) and to introduce the Body of Knowledge as the perspective through which we will investigate the problem space.

The Main Body

The main body contains a structured sweep through the problem space, topic by topic. This is actually quite easy—just follow the Body of Knowledge!

Each topic should list:

- The present situation (description and rating).
- Comparison with the audit baseline (a previous audit). This is also where a compliance rating (score) can be specified.
- Problems with this situation. This is a comparison with the desired situation and a description of any compliance shortfalls.
- Recommended Corrective Actions (CAs).

The main body contains by far the bulk of the report and will undergo a number of revisions long before it is seen by anyone other than the author. It starts out as a series of bullets and random findings recorded during the data collection and evaluation phases. The whole idea is to jot down notes as you discover things. These notes are then re-organised and treated as leads that must be followed up. If the points are satisfactorily dealt with, then you just delete those notes from the report (but keep your notes and findings to refer back to). If they are still a cause for concern as the audit continues, the notes are supplemented with other parts of the topical analysis, introductory text, and CAs to provide context and a more detailed picture. Eventually we have peeled several layers of the onion and have a detailed understanding of the problem space.

Now it is time to re-work the text to show the interactions between topics, and to reflect the state of the system given our new level of understanding. As you can see, this is not a simple, single-pass document, but the result of an iterative development process.

Be sure to re-read this section and re-organise it as appropriate to ensure good document flow and readability. It may be appropriate to split this section into several chapters. For example, you may split up the main body to reflect the top-level organisation of the BOK that was used.

A good principle to use when writing documents of this nature is "SEE": Statement, Explanation, Example. A paragraph should begin with a simple statement. This is then explained and clarified by the following sentences. Finally, to reinforce the point, an example is provided.

This technique can help readers comprehend a large document and will serve to reinforce the major points.

Corrective Actions (CAs)

A CA is a description of how to address the gap between the present situation and the desired situation (compliance). A less useful audit report would merely list all instances of noncompliance. A constructive audit report, by contrast, lists what steps should be taken to bring the system into compliance.

- A CA should not specify technology. Technology is continually changing. The CA should speak in terms of the principles, processes, and controls that must be implemented.

  Wrong: [CA56] Install supermail 15.5.47.

  Right: [CA56] (*Shortfall*) There is no process in place to ensure the currency of relevant software. (*Recommendation*) Implement a process to ensure that the most recent stable version of all key software is installed in a timely manner, with special emphasis on security-related fixes.
- Prioritise each CA. (high/medium/low).
- Index each CA. These CAs should be cross-referenced into a list in the appendices.

The Appendices

The appendices serve to summarise key data from the report, as well as to provide additional information which was not appropriate for the main text of the report.

There will be a number of standard appendices, and you may feel it appropriate to add others. For example, a security audit may use an appendix to list the current vendor patches for the operating systems in question. Such a list would be inappropriate in the main text, but may be of use to readers and so should be included. Should you include an appendix with this sort of time-dependent data, it is essential that it be dated—e.g., "Solaris security patches as at dd/mm/yy." This data would be most relevant if it was as of the date of data collection.

Similarly, the tail end of the document can be a useful place to place explanatory text that may be inappropriate in the body of the report. A major occasion for this use is when you wish to assume a base level of knowledge about a topic, in order to ensure that the main text is not bogged down in tedious explanations. You may be aware that several readers will not have that pre-requisite knowledge, yet it is important they be familiar with some key concepts in order to understand the main text. Such explanatory text is well placed in an appendix. For example, you might provide a short introduction to firewall technologies as an appendix. You can, thus, assume a base level of knowledge in your discussions throughout the text, referring those who aren't familiar with the technology or terminology to the appendix for clarification. The idea here is to use an appendix to contain any information that would detract from the text flow but is a useful inclusion nonetheless.

If you are performing a number of audits, it is tempting to copy and paste these appendices into new audit reports. If you do so, be sure to check them for continuing accuracy and relevance to the present audit.

Each audit report should include a number of standard appendices:

- Appendix A: Corrective Actions, by Priority

  The main body text had corrective actions scattered throughout, each dealing with the topic under examination at that point in the text. These corrective actions should be uniquely numbered, and indexed. This list is then reordered by priority, and presented succinctly in the first appendix.

- Appendix B: Issues Requiring Management Resolution

  In carrying out the audit, you will probably uncover a number of problems that require a strategic resolution. (e.g., the separation of certain services, requiring the implementation of additional core servers). In such cases, you can specify the strategic need, but not the methods by which to achieve them. Similarly, you may uncover a direct conflict between two existing policies.

  This appendix provides a forum for listing and discussing such points of conflict, permitting management to make informed decisions about how to proceed. Remember, the auditor's role is not to decide upon a course of action, but only to make observations and recommendations.
  If you are taking the consulting position, and recommending solutions, it is suggested that you finalise the audit report first, then generate a separate document with any recommendations. Such a document would speak directly to this appendix.

- Appendix C: Issues Identified as Beyond Scope

  During the course of the audit, you may uncover concerns that deal with topics outside the scope of the audit. You should still note these concerns, but in an appendix, not in the main text. These items will have been only partially investigated, as you should not waste resources on investigation once it has become clear that the topic is out of scope. Noting them, however, provides valuable analysis to the sponsor, for little effort.

- Appendix D: Systems and/or Components Examined

  This should be a definitive list of the components examined, the dates they were examined, and any other identifying information. This serves as a reference to the scope of the audit and quantifies that scope.

10. Assessment Criteria

A number of times throughout this booklet I have referred to the need to quantify our assessment. Quantification is the process of *reducing to a measure*. By quantifying our audit, we seek to answer the question, "Did the system pass or fail?"

So the goal is clear, and there are good reasons for it, but how do we actually achieve that goal in a consistent manner?

The answer is to establish some independent rules about the quantification process, then apply these rules consistently across all like audits. To put it another way: consistency of quantification measures implies a defined ratings system. This, in turn, implies defined assessment criteria (i.e., we must define each rating level from worst failure through to best pass, and set the criteria that will indicate each level).

Rating Systems

We can measure a system in a number of different ways, from informal to extremely formal. A good example of an informal but highly useful rating system is taken from Elizabeth Zwicky's tutorial on evaluating a site's maturity.

There, three ratings are defined: Average, Acceptable, and Excellent. (If your site did not rate even an average, it‘s time to panic!)

Elizabeth defines these ratings for several areas. To choose one example, system backups, she suggests the following criteria (I have abridged her suggestions here):

- **Average.** There is a backup system (of some sort) in place; coverage is spotty, and nobody knows exactly how spotty; backups are run by one particular person; tapes are stored in the machine room (somewhere); restores are rare, often don't work, and always generate misery.
- **Acceptable.** People know what is/isn't backed up; most backups are automatic; tapes are stored in a specific place, with some off-site; more than one person understands and looks after the backup systems; restore requests are infrequent, but nobody panics when they occur; partial restores have been tested.
- **Excellent.** An externally supported backup system is in place; a disaster plan exists; separate program-checking backups are getting done; backups and restores are sufficiently automated to be handled by junior personnel; backups are routinely tested; backups are available for every computing platform; a system exists for backing up home machines and laptops.

This rating system provides a set of meaningful and highly practical lines in the sand, and allows a person to rate the maturity of their site's practices. The criteria must be defined for each evaluation point for each subsystem, but someone experienced with the subject matter should be able to make meaningful ratings using such a technique.

Measuring Compliance

Towards the more formal end of rating systems is a subject-matter–independent method for rating compliance with policy. A mature IT organisation should be driven by policy in each area, and so auditing compliance with policy is equivalent to auditing the organisation's *capability maturity*, that is, the maturity of its capabilities in that area.

For a rating system to be of most value its rules must be independent of the actual assessment criteria, i.e., they should be related, not to the technology of the system, but rather to independent quality standards. After all, we are assessing a system's conformance to a specified standard of practice, and so should quantify the audit in terms of this conformance.

In other words, what we are rating is the *degree of compliance*.

Below is a suggested rating system that can be used as a starting point for developing your own relevant rating system:

Coarse	*Fine*	*Rating*	*Guidelines*
0–1	00–19	Poor	No ability/policy/procedures Ineffective results
1–2	20–39	Weak	Partial ability/policy/procedures Fragmented usage Inconsistent results
2–3	40–59	Marginal	Usage in major areas Consistent positive results
3–4	60–79	Fair	Adherence to procedures Usage in most areas Positive measurable results
4–5	80–99	Qualified	Adequate procedures Practice is integral part of process Positive long-term results
5	100	Outstanding	Excellence in practice well recognised Consistent long-term use Consistent world-class results

This system represents an additional layer of abstraction over the previous one presented. The BOK should define what policies and standards must exist, and these criteria can then be applied to measure the site's compliance with them.

Categories and Weightings

For an audit to be of most use, it should be organised into groupings of related topics. As we have already seen, the audit should be a topical analysis of the problem space. Such an analysis will result in a primary dissection into a small number of top-level audit areas right from the outset. Immediately we can now assess the system, area by area, and rate each area independently, rather than just providing an overall pass/fail. Any repair effort will normally be made as a series of discrete projects, and this breakdown into areas allows us quickly to identify the areas most in need of attention.

As you will see from the SA-BOK checklist (for details, see Appendix A), important categories in system management include:

- Change management
- Facilities management
- Problem management
- Asset management

Some of these areas will have a more significant impact on the correct operation of the system as a whole than others. We should weight each topic area so that its relative importance will be properly reflected in the overall assessment results. For instance, change management will have a far greater impact on production quality than asset management will. Both are important, of course, but the ability to track asset movements accurately is not as likely to contribute to system availability as will good change management procedures.

Whatever the system, it is important to take weightings into account when drawing up a topical audit checklist.

This same acknowledgment of varying levels of importance should continue into the breakdown of topics into subtopics and so forth. Just as change management is more significant than asset management, so too, the existence of a multi-tiered change management process is more significant than having automated account maintenance commands.

Showstoppers

Another fact of life is that certain aspects of a topic represent *showstoppers*—the automatic failure of that topic, category, or even the entire audit. A lack of clear policy, to take a prime example, results in an automatic failure. Without policy, we have no benchmark against which to judge the rest of the audit.

Even when you encounter a showstopper, it is important to continue with as much of the remainder of the audit as is reasonable. To stop at the first fault may mean that fifty audits are required before a pass can be obtained. It would be far more productive to identify the fifty problems in the first audit, so that all of them could be addressed before re-auditing.

You will find that most showstoppers will not, in fact, stop the show at all. Whilst the topic may immediately fail, we can proceed with the audit by making certain reasonable assumptions and stating them explicitly. Where a lack of policy is the problem (an all-too-frequent occurrence), then we can use our knowledge of reasonable practices (the industry benchmark practices) as the basis for the audit.

11. Controlled Improvement Programmes

The act of auditing as an isolated activity is a negative, critical exercise. The result is a list of things that are "wrong," as measured against some criteria. No audit (except perhaps for a blame allocation exercise) is intended to be performed in such isolation. Each audit should make specific recommendations as to how to improve the system to a pass grade when re-assessed against the same criteria.

It is also a passive exercise. No changes should have been made as part of the audit process. (The system will, however, probably have undergone changes during the time since the audit commenced.)

The question then arises, "How do we complete the job?" The systems will often be mission-critical production environments, and their user communities will not take kindly to anything that reduces their system reliability or availability, even if the end result will be an improvement in those same measures. Appropriate care must be taken in the planning and execution of the recommended actions to minimise impact on the user population.

These technical improvement works are discrete projects, and should be treated as such. This is important for many reasons. Each project must be clearly allocated the resources necessary to reach successful conclusion. Each should provide systems personnel with a sense of purpose during the project and a sense of achievement when it is completed. Progress should be tracked against the project plan, and the results re-assessed against the stated objectives at the project's conclusion.

Controlled improvement programmes should all take a similar basic approach:

1. **S**tudy and strategise. Determine and recommend the direction forward.
2. **P**lanning. Prepare a formal (detailed) project plan.
3. **A**uthorisation. Obtain management approval to proceed with one or more improvement projects.
4. **C**ontrolled repair. Execute the controlled repair project.
5. **E**valuation. Re-assess and review.

Step 1—Study

1.1 Study the audit report.

1.2 Draw up strategies and proposals—the SOPPADAR approach (see below).

1.3 Obtain management decisions regarding expenditure, technology directions, and other alternatives.

The result of the audit report is the identification of a number of defects—places where the present situation has failed to comply with the required standards. The objective of any improvement programme, therefore, must be to address these defects in order to bring the system into compliance.

As we have stated already, there is no One True Way™. There will always be a number of ways to address any problem, each with associated pros and cons. It is not the job of the auditor to choose among the alternatives. The auditor's responsibility extends only to recommending corrective actions to bring the system into compliance.

It is, however, your job as the improvements project leader to ensure that informed decisions are made by management and that reasonable progress can be made towards addressing the shortfalls identified.

Study the Report

The first step is, of course, to study the audit report and understand its recommendations, as well as the Body of Knowledge used during compilation of the report. As with any report, you do not have to accept the audit's contents as incontrovertible fact, but you will need to justify any disagreements you may have with it, just as the audit report must justify its conclusions.

Prepare Proposals

Where strategic expenditure has been identified in the audit report, decisions must be made in order to make progress. For example, the audit report may call for the redistribution of host services. Expenditure on a variety of alternate products may also be required. Similarly, a myriad of other technology decisions must be made before progress can commence.

In order to reach a decision that is appropriate for the organisation, the manager who is responsible for that decision must rely on staff members to supply sufficient information to allow an informed, thoughtful decision to be made. This is the classic cost-benefit analysis.

The most appropriate format for such analyses is the SOPPADAR:

> **Subject.** What is the subject of this discussion paper?
>
> **Object.** What are our objectives? The answer should match one or more CAs from the audit report.
>
> **Present situation.** Describe the existing situation. Paraphrase or expand on the audit report as appropriate.
>
> There will be a number of ways to meet the objectives from the present situation. For each of these, you should present:
>
> > **Proposal.** Summarise the proposed path and its important aspects, such as cost, resource levels, risk, etc.
> >
> > **Advantages.** What are the advantages of choosing this proposal?
> >
> > **Disadvantages.** And the disadvantages? Disadvantages may include its cost, implementation time, restrictiveness, implied dependencies, etc.

Actions. What must be done to move forward on this proposal if it is chosen?

Recommendations. Having presented each major alternative, you may evaluate the advantages and disadvantages of each proposal and make a recommendation of which (one or more) should be approved. You may consider that the advantages in one particular proposal significantly outweigh its disadvantages. There is nothing wrong with stating your opinion.

The SOPPADAR is a standard technique for writing proposals where a choice of direction exists and management must assume the responsibility for making that choice. It is the precursor to a formal decision-making step.

Remember, it is management's prerogative to choose which proposals (if any) they will accept. It is the job of management to weigh the risks of committing to a particular path versus not going down that path. Management must accept, avoid, or assign that risk. (See Chapter 2 for more on this.)

Step 2—Plan

Having identified those defects which must be addressed and having obtained management's decision as to how to address these defects strategically, we can now proceed to determine how to meet these objectives in terms of the available technology. In other words, it is now time to draw up a detailed project plan.

Our improvement programme is all about pro-actively applying structural changes to a subsystem with a minimum of disruption to the user community. We must do what we can, therefore, to minimise the number and/or duration of outages and maximise the warning we give to the user base of that downtime. The key attribute of this project plan is *predictability*.

The task of drawing up a project plan involves translating the *topical* analysis presented in the audit report into a *temporal* project plan.

If the improvement works are non-trivial, we need to plan them carefully in order to meet the goals of minimum disruption and predictability.

Project Plan = Alternatives + Decisions + Priorities

As a rule of thumb, anything that requires more than a few days of effort should be subjected to a formal project plan. The plan's complexity should reflect the complexity of the underlying task.

Step 3—Authorisation

Once a full project plan has been developed, you will need to get management sign-off. The resulting improvement project will be non-trivial. It will have an impact on the user community, it will draw on the organisation's resources, and it may even entail resources from outside the organisation. It is essential, therefore, that management approve this project.

Step 4—Controlled Repair

Go to it!

Step 5—Evaluate (Re-Audit and Review)

Once the project has been completed, we should re-audit the system, or at least all affected subsystems, and evaluate the success of the project.

Do not underestimate the value of this last step. Management made a decision to apply time, money, and resources to an improvement project. They need to be shown that they received adequate value for that investment. This vastly increases the likelihood that other, possibly related projects will be given permission to proceed.

As well as re-auditing, which is used to gauge progress, we should now review the project and the overall process. This re-audit and review are essential aspects of total quality management. In the review we will seek to determine whether the project could have been managed better, review what problems were encountered and what we can learn from them, and determine whether the process needs further refinement.

Appendix A. System Inspection Checklists

One of the important messages of this booklet is that an integral part of the audit process is the development of an appropriate framework for analysing the problem space. This may be called a *methodology*, *checklist*, *framework*, or *body of knowledge*.

Several such works exist which relate to the IT field in various forms. These include:

1. **AS/NZS-4444.** Australian Standard for Information Security Management. Based upon BS-7799. *<http://www.standards.org.au>*

   This extensive and well-organised standard covers most system management functions as they relate to information security. It uses the wider definition of security, including confidentiality, integrity, and availability. Areas such as business continuity planning, personnel security, physical and environmental security, and computer and network management are examined and controls are suggested.

2. **ISACA COBIT.** Information Systems Audit and Control Association, Control OBjectives for Information and Related Technology. *<http://www.isaca.org>*

   This high-level process-and-control–oriented set of 34 objectives is divided into four domains: planning and organisation, acquisition and implementation, delivery and support, and monitoring. A major focus of these controls is to ensure traceability from the business drivers through to the implementation. The COBIT mission is "to research, develop, publicise 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."

3. **SA-BOK.** The Systems Administration Body of Knowledge. *<http://www.sysadmin.com.au>*

   This work is an attempt to define the field of systems administration in terms of *key areas of responsibility* (KARs), and to examine the controls necessary for varying levels of IT maturity in each of these areas. By using the KARs as the basic organising principle, the SA-BOK seeks to help bridge the communications gaps with other communities. The KARs presently identified include change management, problem management, production management, asset management, facilities management, network management, server management, software management, data management, data security, business continuity planning, performance management, process automation, capacity planning, technology planning, and service management.

The SA-BOK project, headed by the author of the present booklet, is attempting to leverage the skills and knowledge of the SAGE community world-wide.

Appendix B. Audit Resources

1. **Information Systems Audit and Control Association** (ISACA). ISACA is a membership organisation which developed the COBIT audit methodology and framework. <*http://www.isaca.org*>

   Certified Information Systems Auditor (CISA). The certification awarded by ISACA.

2. **International Information Systems Security Certification Consortium** (ISC)2. (ISC)2 is a certification authority. <*http://www.isc2.org*>

   Certified Information Systems Security Professional (CISSP). Certification programme run by (ISC)2.

3. **Information Systems Security Association** (ISSA). ISSA is a membership organisation. <*http://www.issa-intl.org*>

4. **Computer Security Institute** (CSI). CSI is a membership organisation. CSI runs training programmes and workshops and has developed the Information Protection Assessment Kit (IPAK). <*http://www.gocsi.com*>

Bibliography

[Fa96] Farmer, Dan, and Wietse Venema. 1996. Security Auditing Workshop. <*http://www.fish.com/security/auditing_course/*>

[Ga96] Garfinkel, Simson, and Gene Spafford. 1996. *Practical Unix & Internet Security.* 2nd ed. O'Reilly & Associates. ISBN 1-56592-148-8.

[Gr93] Grottola, Michael G. 1993. *The UNIX Audit: Using UNIX to Audit UNIX.* McGraw-Hill. ISBN 0-07-025127-4.

[Li97] Lirov, Yuval, et al. 1997. *Mission-Critical Systems Management.* Prentice Hall. ISBN 0-13-240292-0.