Using Assurance Models in IT Audit Engagements

Adrian Baldwin, Yolanta Beres, Simon Shiu
Trusted Systems Laboratory
HP Laboratories Bristol
HPL-2006-148R1
January 29, 2008*

audit, assurance, compliance, Sarbanes-Oxley, SOX, risk, security

The document describes an innovative way to assess the effectiveness of internal IT controls where the control framework is first captured in the models and then the models are used to analyse the evidence gathered from the IT environment. The aim is to lift the risk and control management lifecycle from a series of people based processes to one where model based technology enhances, connects and where appropriate automates the process. Modelling in such an approach means capturing the relationship between controls and the way the controls should be analyzed for effectiveness and compliance to regulations and internal policies.

This document presents how the model based assurance approach has been applied to automate the analysis of critical IT internal controls during several IT application audits in HP, and the value and benefits we have seen in using models to drive real-time analysis and measurements of the operating environment.

Internal Accession Date Only Approved for External Publication

© Copyright 2006 Hewlett-Packard Development Company, L.P.

Using Assurance Models in IT Audit Engagements

Adrian Baldwin, Yolanta Beres, Simon Shiu[1]

Trusted Systems Laboratory

HP Laboratories, Bristol, UK

Abstract

The document describes an innovative way to assess the effectiveness of internal IT controls where the control framework is first captured in the models and then the models are used to analyse the evidence gathered from the IT environment. The aim is to lift the risk and control management lifecycle from a series of people based processes to one where model based technology enhances, connects and where appropriate automates the process. Modelling in such an approach means capturing the relationship between controls and the way the controls should be analyzed for effectiveness and compliance to regulations and internal policies.

This document presents how the model based assurance approach has been applied to automate the analysis of critical IT internal controls during several IT application audits in HP, and the value and benefits we have seen in using models to drive real-time analysis and measurements of the operating environment.

1 Introduction

New regulations and constant risk of information-security threats are forcing organizations to more vigorously examine effectiveness of their internal IT controls and processes. To deal with these pressures, organizations are calling auditors to make sure their systems comply with corporate security policies and to implement appropriate internal controls that mitigate the security risks to their critical information, applications and infrastructure.

Internal control is broadly defined as a process put into effect by management and other personnel, designed to provide reasonable assurance regarding the achievement of objectives in the following overlapping categories:

- Effectiveness and efficiency of operations;
- Reliability of financial reporting;
- Compliance with applicable laws and regulations.

For a control to be effective, actual results must be compared to expected results or standards, and corrective action must be taken when indicated. The identification of the effectiveness of the mitigating controls usually falls into the responsibility of internal and external corporate auditors. Auditors

[1] The authors would like to acknowledge the contributions and support of Francisco Montes, Frederick Brown, Angela Davis-Brewer and Steve Stein from HP-Internal Audit without whom the application audit pilots would not have been possible.

have developed various methodologies to assess compliance to Sarbanes-Oxley Act (SOX) [2] and other regulations but most of them are still very time consuming and labour intensive. In the world of growing regulatory mandates and industry-based requirements where besides SOX [1] organizations have to meet other regulations such as HIPPA [10] and PCI [9], to name a few, auditors are faced with constant pressure to provide more timely and ongoing assurance that controls are working effectively and risk is being managed.

Model based assurance approach presented in this document aims to model the control framework and use the models to systematically analyse the evidence on the effectiveness of the implemented controls. This immediately lifts the assurance lifecycle from a series of people based processes (risk management, control design and implementation, audit and review) to one where model based technology enhances, connects and where appropriate automates the process. This document, in particular, examines how our approach can be used to support part of auditor's lifecycle of listing the control sets, defining the required test work, gathering and analysis of evidence and identification of risks. In different exercise we have also explored how an assurance model can represent the control architecture and its relationship to the enterprise architecture and how it can be used to support an enterprise risk life-cycle [11].

This document presents the results of the pilot and investigation into the extent to which the auditing process can be captured in models and automated. Automated analysis approach allows auditors to dedicate more time to the assessment of risks and the adequacy of controls, rather than manually examining the evidence. It can also allow auditors to deliver timelier and higher-quality results. And, it can help audit management allocate precious — and scarce — staff resources better to focus on high risk or significant areas of exposure to the organization.

The result of the pilot was to provide evidence of the value of the model based approach in a particular and highly relevant context. In addition, by describing the process for creating models, the document shows the extent to which the proposed modelling approach can transform part of the yearly auditing process into a continuous auditing. Such a transformation goes beyond simplifying the auditing process; it changes the nature of this process, transforming it from data analysis and assessment of deviation into a real-time monitoring and continuous compliance.

The document is organised as follows, section 2 provides a short overview of the architectural components of model based assurance framework and a high level description of the lifecycle of creating model based reports. Section 3 describes the requirements of the gathered from HP auditors and the process of developing models that capture and meet those needs. Section 4 describes in detail the models that were deployed to analyse security controls in IT application audit pilot. It summarises the results and benefits, and discusses the implications for continuous compliance. Section 5 discusses current and future work of developing and deploying the models for analysis of the database and infrastructure controls. Section 6 draws final conclusions.

2 Model-based Assurance Framework

The main concern of model-based assurance approach is providing assurance that a system, application or service is 'fit for purpose' and 'well run'. The tools and vision behind this approach have been described in [3], [4], and the overall architecture is shown in figure 1.

Implementation of model based assurance approach consists of two phases: (1) developing a model for a given IT environment (applications or infrastructure) based on control frameworks; (2) using the model to instrument and analyse system level data and produce a report comparing the way the system (including the control environment) is running against the description in the model.

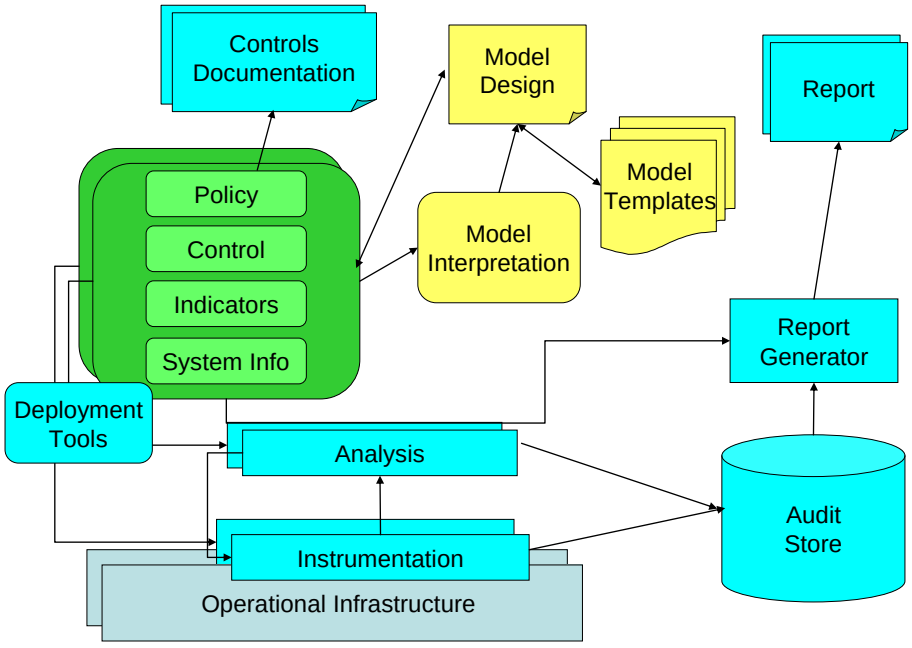


Figure 1 Model Based Assurance Framework

2.1 Model Development

The assurance model in our framework is a set of connected nodes each defining a specific data analysis type or data combination rule. The nodes are connected among themselves and also to the various data source types to form an acyclic graph structure as can be seen in the examples in later sections such as figure 4.

Each node represents an entity within the control framework. The entity can be a control, risk, a detailed test-work of a control or an indicative metric. Connections between the nodes represent relationships between these entities with the meaning being dependent on the selected input and output data sources in the nodes.

The node has a descriptive element including a type and a set of attribute value pairs that describe what the node is modelling. The node also has rules that define the relationships between the inputs and outputs, or more

specifically how outputs are derived from a set of input data sources. These rules define the type of the node and differ for each node type according to the number of input sources used and how these input sources are manipulated to produce resulting outputs. An array of different analysis node types are available for building the model; for comparing various sets of data, performing checks such as separation of duty, checking event orders etc.

When combining metrics, or evaluating the results from testing, certain types of nodes can be used to define thresholds on specific data values that result in different colour flags in the final report. Other type of nodes can also be used to specify how the outputs from the various connecting nodes should be combined.

To create and visualise the models we have created a model design tool as part of the framework that allows for graphically drawing out the relationships between different parts of the model. Using graphical modelling approach the model is represented as an acyclic graph structure that links various graph nodes. New models can be built to relate various controls to high-level policies as well as to detail tests. Some new models may just require customising a set of templates, for example the ones that have been already developed by us to test security controls for compliance to Sarbanes-Oxley requirements.

Once the model has been created, the tool renders the model in an xml structure suitable for our analysis engine. The resultant model drives both the analysis engine and the reporting system the final results of which show whether the controls are being run in line with expectations.

2.2 Model-based Analysis and Data collection

During the continuous analysis and reporting stage, the analysis system takes the model description and deploys it as a set of analysis objects corresponding to nodes in the model that analyse the system information or other analysis results. The model defines both the flow of information between these objects as well as the details for each analysis object.

Some of the analysis components in the model, mainly at the lower leaf levels, define what raw information has to be gleaned from the IT environment. At the system deployment stage a step then needs to be carried out to determine how best to collect this raw information.

Certain information may be available directly through existing monitoring agents, log files or databases. In other cases the IT environment may require additional instrumentation in order to collect the required information. Rather than work on a collection framework we have assumed that information can be made available from a number of sources via a centralized database.

3 Assurance Models and IT SOX Audits

In the context of SOX compliance, we have been working with IT auditors to test how our assurance models can be used to capture risks and associated controls that are being assessed by IT auditors for SOX compliance. In addition we have been piloting deployment of model-based analysis on top of data being continuously collected from real IT applications and systems. The

aim of this work is to provide continual analysis of critical IT controls and so reduce the burden of manual assessment and testing.

Usually the sheer volume of transactional data and potential metrics makes data analysis difficult and leads to data overload. Our aim was to capture in the model only the key controls, and so minimise the amount of data that have to be gathered. A key to success is relying heavily on process-based analysis and metrics rather than just transactional and event-based data.

HP's internal IT auditors have already been active in trying to address the need for continuous monitoring of key IT controls. To achieve that they have been pioneering the use of key lagging indicators that can be continuously gathered from IT environment to indicate potential issues with IT controls and hence predict the likely outcome of an audit. With the model-based approach we aimed to both capture their current practise of collecting the key metrics but also extend it to include more complex tests and analysis of key controls.

In designing an assurance model for SOX compliance we tried to capture best practices that HP's IT internal auditors use in assessing risks and associated IT controls. Undoubtedly, different auditors are likely to take different approaches. Nevertheless we expect the resultant model is indicative and representative of the audit process in a SOX compliance domain.

We have been applying and testing our model based assurance approach in two main contexts of IT audits: application audits and infrastructure audits. The application audits cover most of IT audits, at around 70%. The infrastructure audits are usually data centre audits that cover both the servers and in most cases databases.

The testing of our approach has been most vigorously applied within the context of application audits and SAP applications, in particular. The next section will describe in detail the models used and the results of our pilot in application auditing space.

4 IT Applications Audits: Pilot for Security Controls

As an input for creating the model and necessary analysis components we have used IT controls matrix used by IT auditors for application audits. This lists controls that are tested as part of SOX compliance audits, also giving indication of the risks the application is exposed to if these controls are not working properly. An example of controls matrix is shown in figure 2. The lists of audited controls is usually dictated by following the guidelines from audit governing bodies of perceived risks to specific business processes [2] and by applying best practices of what IT controls are critical to manage these risks such as dictated by CoBIT, ITIL, and COSO [5,6,7].

Each control area from the control matrix is then captured in our model. In this particular case the model covered five separate control areas:

(1) account termination control status

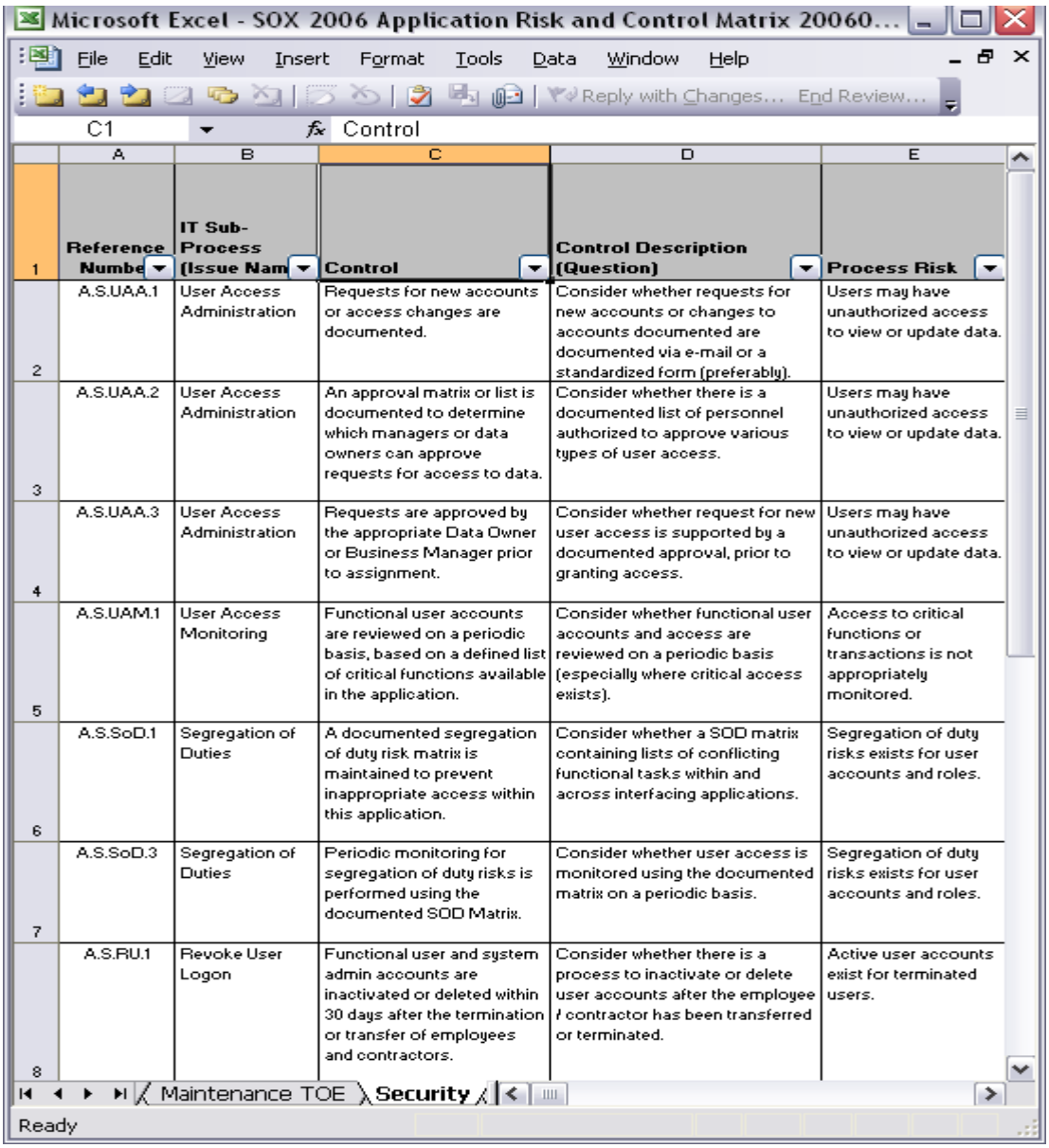
(2) new user account request process

(3) appropriateness of user accounts authorization and access

(4) account usage indicators

(5) segregation of duty conflict analysis.

In each control area, based on the documentation and on consultations with the auditors, we then selected a set of analysis modes and metrics that would indicate how the control is working. We will look at each of the 5 areas in more detail in the next sections.



Reference Numbe	IT Sub-Process (Issue Nam	Control	Control Description (Question)	Process Risk
A.S.UAA.1	User Access Administration	Requests for new accounts or access changes are documented.	Consider whether requests for new accounts or changes to accounts documented are documented via e-mail or a standardized form (preferably).	Users may have unauthorized access to view or update data.
A.S.UAA.2	User Access Administration	An approval matrix or list is documented to determine which managers or data owners can approve requests for access to data.	Consider whether there is a documented list of personnel authorized to approve various types of user access.	Users may have unauthorized access to view or update data.
A.S.UAA.3	User Access Administration	Requests are approved by the appropriate Data Owner or Business Manager prior to assignment.	Consider whether request for new user access is supported by a documented approval, prior to granting access.	Users may have unauthorized access to view or update data.
A.S.UAM.1	User Access Monitoring	Functional user accounts are reviewed on a periodic basis, based on a defined list of critical functions available in the application.	Consider whether functional user accounts and access are reviewed on a periodic basis (especially where critical access exists).	Access to critical functions or transactions is not appropriately monitored.
A.S.SoD.1	Segregation of Duties	A documented segregation of duty risk matrix is maintained to prevent inappropriate access within this application.	Consider whether a SOD matrix containing lists of conflicting functional tasks within and across interfacing applications.	Segregation of duty risks exists for user accounts and roles.
A.S.SoD.3	Segregation of Duties	Periodic monitoring for segregation of duty risks is performed using the documented SOD Matrix.	Consider whether user access is monitored using the documented matrix on a periodic basis.	Segregation of duty risks exists for user accounts and roles.
A.S.RU.1	Revoke User Logon	Functional user and system admin accounts are inactivated or deleted within 30 days after the termination or transfer of employees and contractors.	Consider whether there is a process to inactivate or delete user accounts after the employee / contractor has been transferred or terminated.	Active user accounts exist for terminated users.

Figure 2 SOX Risk and Control Matrix for Security

To perform the tests and generate the indicators based on the model we created collectors to gather data from various data sources. To get the application data which in this case was SAP we have created regular pull from the centralised database that holds user data for 25 critical SAP applications in HP. In addition to the application data that included user account data as well as user role data our collectors also gathered data from: (1) Enterprise Directory (ED) on a daily basis about active and terminated employees, (2)

account approval email archives on a monthly (or as required) basis (in cases when the account provisioning is automated). All this data has been pulled into a centralized database. Figure 3 shows the architecture of the overall solution.

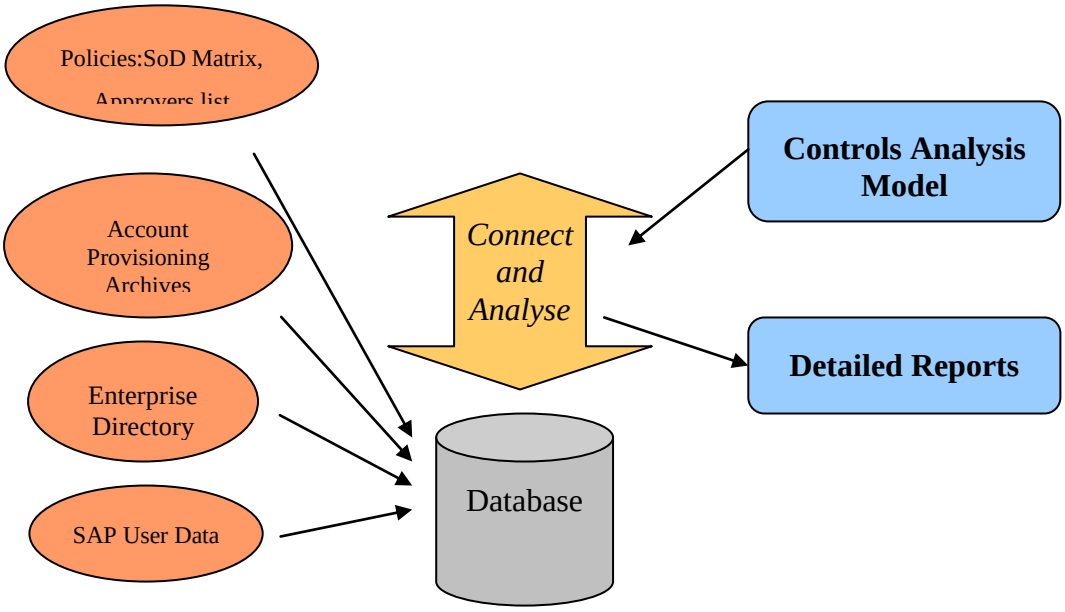


Figure 3 Architecture of the overall solution for analysis of IT application security controls.

4.1 Revoke User Logon/ Account Termination

Account termination control deals with the mitigation of the risk of existence of active user accounts for terminated and transferred users. It is assumed that this control is working if the following condition is met:

> *Functional user and system admin accounts are inactivated or deleted within 30 days after the termination or transfer of employees and contractors.*

To see how this control is working an auditor performs one major test: taking all active users accounts on application he checks how many of them have been terminated as employees for more than 30 days. This manual analysis requires first downloading a system user list from an application and the active employees list from ED into a spreadsheet. If the users are found on the application or system that are not active employees, then these employees are matched to the ED terminated list to get actual termination dates. The comparisons between the lists are done using query based tools, one of them being Audit Command Language (ACL) [8] (used 50% of the time). The size of this data is often large; it takes around 1-2 hours to perform this analysis depending on the application and organisation size.

To automate this manual analysis, we have created an analysis model that replicates exactly the tests being performed by the auditors[2]. In addition to looking at the terminated employees, we also generate indicators about expired accounts. Figure 4 shows the breakdown of account termination analysis.

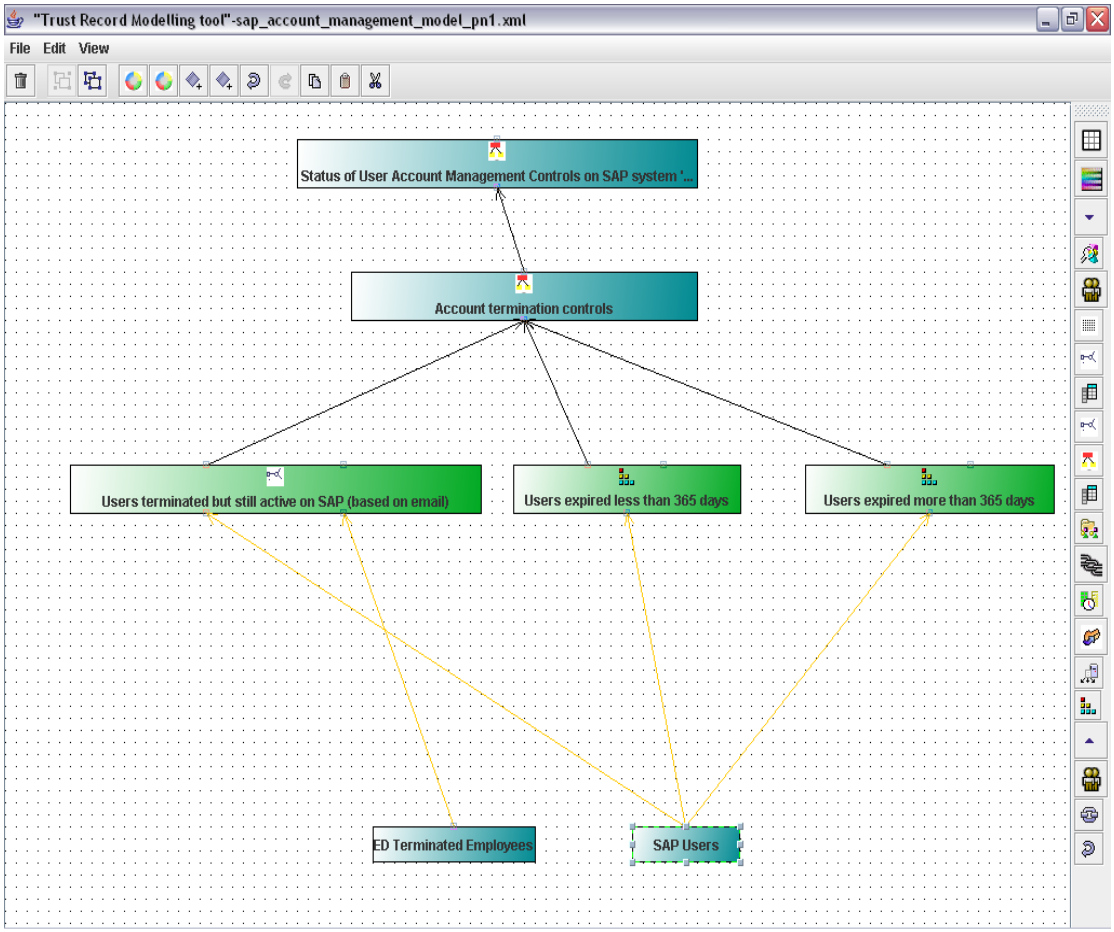


Figure 4 Model for testing account termination control

To perform terminated employee analysis we include in the model an analysis node that is capable of looking across two data sources and excluding data that is not found in one data source based on comparing one unique field and a date.

Most organisations have an ldap-based Enterprise Directory (ED) where the status of each employee is recorded, such as start and termination dates together with the individual email and a unique identification number. We use this data as one data source for the analysis and the user data from the application as another. The threshold on the results is set in such way that if

[2] During this pilot the model was created to test for terminated employees only. Another aspect of this control is transferred employees, which has not been captured in the model described below.

at least one active application user is found that has been terminated in ED as an employee, the red flag is raised.

The metrics about the expired accounts will be created by using one type of analysis node "*CountEntries*" that as the name suggests counts unique entries that meet specific conditions. In this case we use two different conditions: accounts that have been expired for less than 365 days, and accounts that have been expired for more than 365 days. Often according to security policy in place the accounts expired for more than 365 days are to be removed from the system. High number of expired accounts still existent on the system shows that the process of removing such accounts is not working.

The analysis based on the model above takes around 15 min to complete and the results can be viewed in a dashboard style as in figure 5, and also in further detail for the problematic accounts as in figure 6.

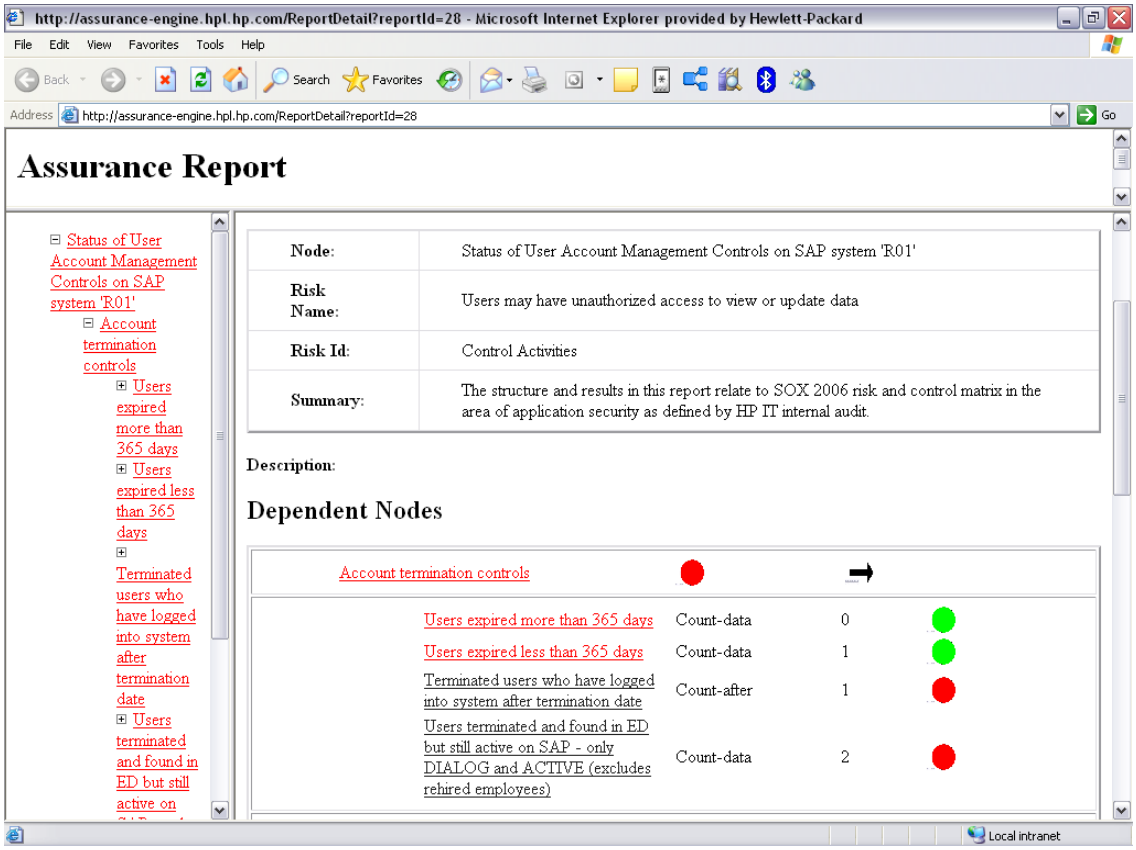


Figure 5 Dashboard style results of account termination control analysis

Providing not just dashboard reports but also detailed reports that can be exported into spreadsheets proved very important for auditors, as they need to document and capture results in the final audit result documents, as well as in presentations.

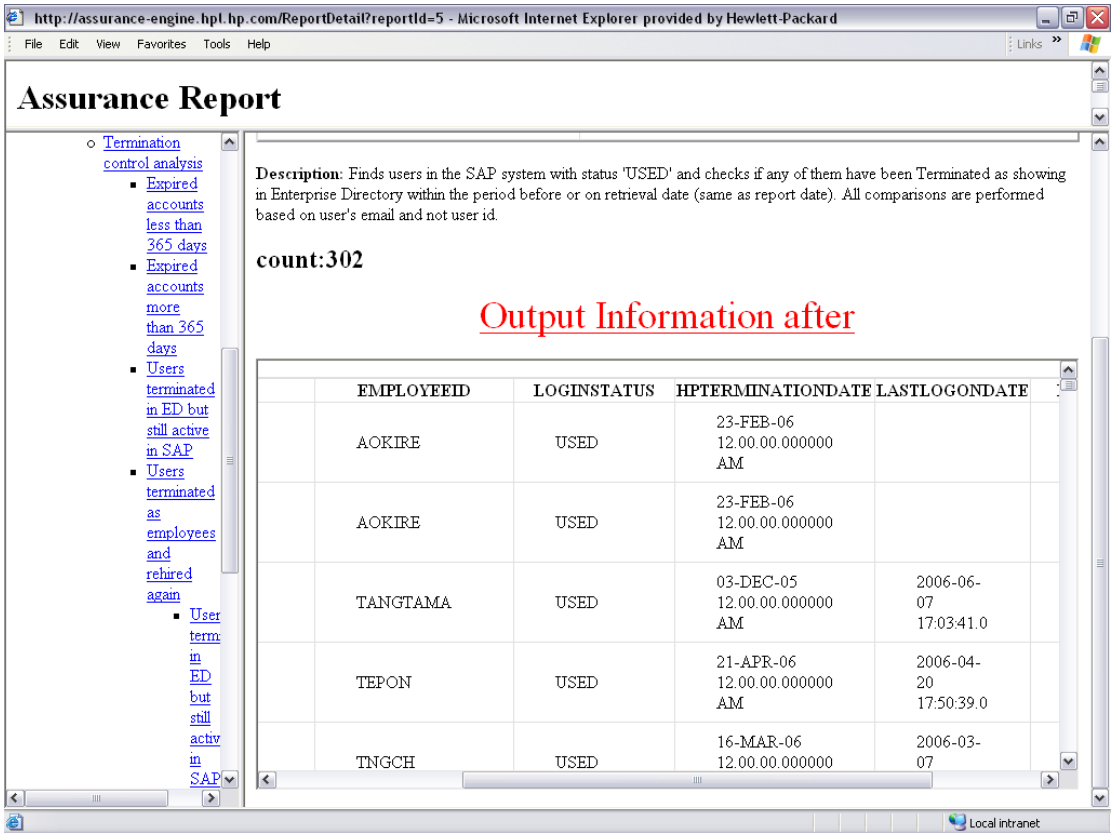


Figure 6 Detailed results of account termination control analysis

4.2 New Access Approval Controls

The next important control area is concerned with mitigation of risk of users having unauthorized access to view or update data. It encompasses 2-3 specific controls, all around the controls for new access request and request approval process:

- *Requests for new accounts or access changes are documented.*
- *An approval matrix or list is documented to determine which managers or data owners can approve requests for access to data.*
- *Requests are approved by the appropriate Data Owner or Business Manager prior to assignment.*

Based on the above control description and the actual tests performed by the auditors during the application audits, we have selected an extensive set of analysis nodes and automated tests to be captured in our model. All of them are shown in figure 7.

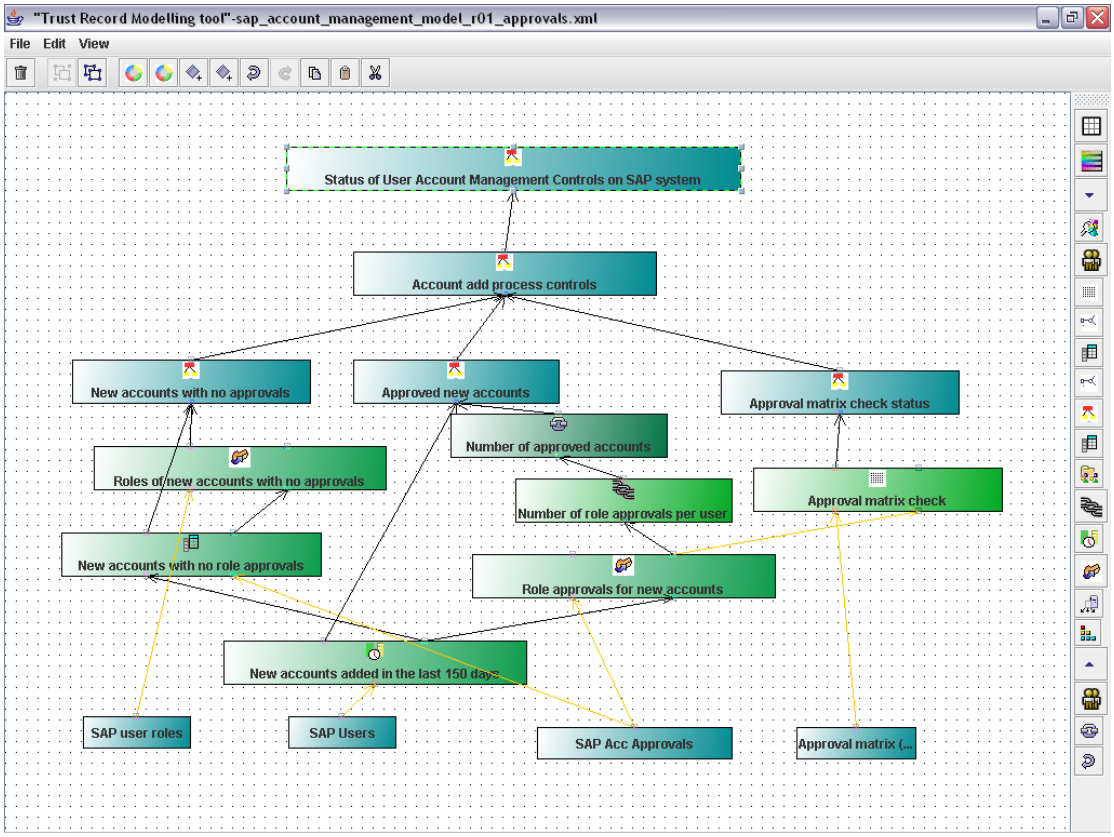


Figure 7 Model of new accounts authorisation control

Within the model we first use a node that selects new accounts that have been added within the specific period (in the example shown we used 150 days period but the size can be increased based on audit requirements); the application's user data is used for that.

For the next two types of analysis the aim is to find details of the new added accounts that have no approvals and that have one or multiple role approvals together with the actual details of each approval. As the data source for the approvals we use the email approval archives that are being generated on a regular basis by the application management team[3].

For accounts with no approvals we also check how many actual roles/authorisations are assigned to those accounts. This is important for understanding how big is the risk of a particular account being created without any approvals. If an account has been created but has no roles or authorisations assigned the risk might be minimal. On the other hand, if an account has many authorisations and none of them have been approved, the risk that this user has performed privileged transactions without authorisation is much bigger. Even if most account assignments can be justified by the

[3] The model works well when the application follows this standard approval process. We have observed this type of archive generation process across four SAP applications in HP. However, it might be not be consistent across all the applications. In those cases an investigation needs to be carried out how to obtain the approval information; if archives are not kept one option would be to instrument the process to generate the approval events every time an account has been added or changed.

management team, the high number of such exceptions would indicate that the process of approving new accounts is not adequate and thus has to be changed to minimise exceptions.

The results of the analysis of accounts with approvals would include details of individual approval such as role/authorisation approved, details of the approver, time approval received and so on. These details are then used for an approval matrix analysis to check that role requests have been approved by the appropriate Data Owner or Business Manager as defined in the approval matrix (if such is available). The results of this analysis are sets of "correct" approvals (approvals by the data owner as specified in the approvers matrix) and of "incorrect" approvals (approvals by the person not registered as the data owner). These results are again used to determine the risk of somebody gaining improperly authorised access and also to evaluate if the approvals process in place is adequate.

During the iterations with IT auditors at the actual application audits we found that automated analysis of new accounts controls are extremely valuable as it considerably minimises the auditing time and effort. It also allows for higher penetration and coverage.

The current audit practise when testing this particular control is to select a limited number of samples that cover the entire period, usually from 6 months to a year. For the selected sample user accounts the control is then fully tested. Sampling is based on statistical theory and is universally accepted as an appropriate testing approach. If testing all new accounts without sampling an auditor would need to examine a very large number of evidence that would take weeks[4]. In addition to sampling auditors also review the process itself, which gives insight into how exceptions, etc. are defined and handled.

Such an approach has several limitations, though. Since sample accounts are selected randomly, the final sample might at the end include only accounts for which the control has worked effectively. Such an approach will not necessarily show if during the certain period there wasn't any risks of unauthorised accounts present on the system. It also does not give a full view of how the exceptions were actually handled. To better estimate the risk of unauthorised access on the system it would be beneficial for an auditor to examine all new account approvals and all account changes as is possible using automated approach. Based on the results of this analysis, further investigation can be carried out only into the problems that were found.

The results of applying this model to the analysis of data gathered from email archives containing account approvals, as well as from user account information such as user roles, creation date and so on is shown in figure 8.

[4] For an application that has 2000-3000 users there are around 600 new accounts per half year with an individual account having 20 or more roles, resulting in 2000-3000 role approvals/denies.

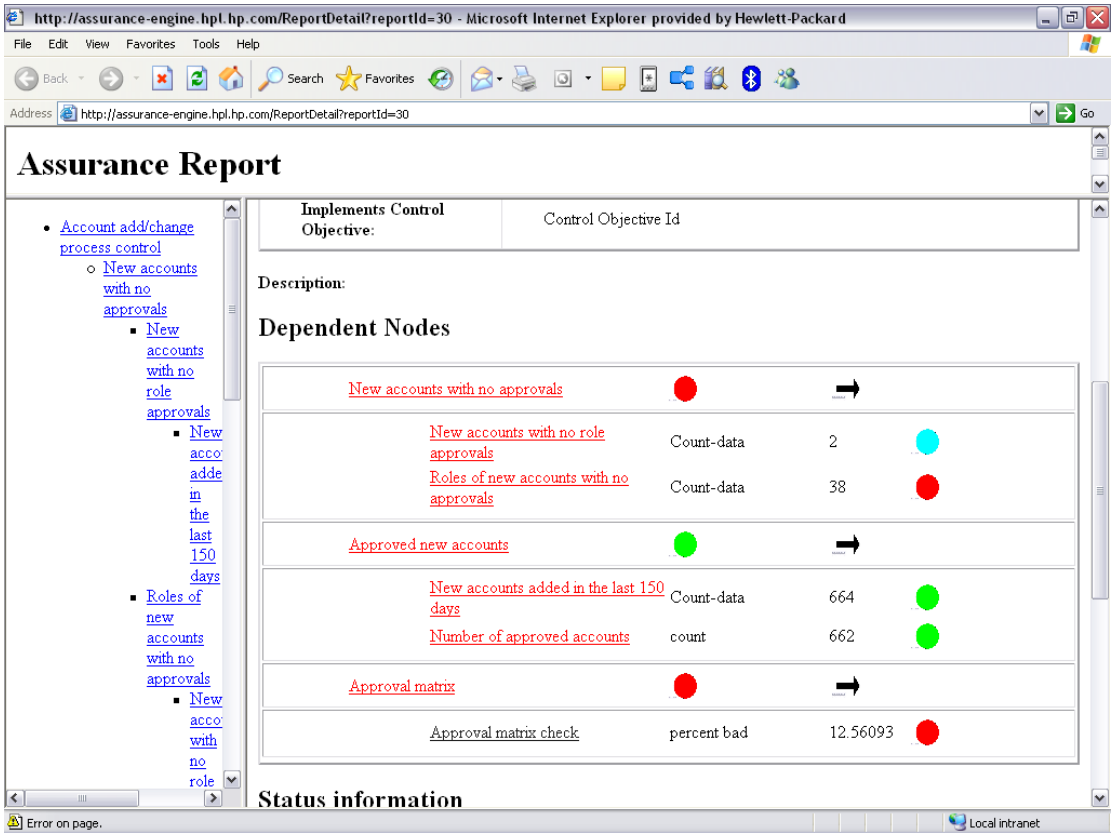


Figure 8 Results of account approval control analysis.

4.3 Correctness of Accounts and Account Usage Indicators

The results of the above analysis have to be viewed in the light of how good the accounts are managed overall and also in the light of how often the application is being used.

To present this view we have created some additional analysis and metric gathering that though not part of any specific control area are indicative of how application is managed overall.

In the area of account correctness we aimed to examine two things:

1. If all accounts on the system have an associated human person that can be traced to an employee number in the Enterprise Directory.
2. If no duplicate accounts have been created for the same person.

The account correctness is necessary in order to correctly trace the accountability in case of risk exposure or incidents. The model that includes analysis for both the above tests is shown in figure 9.

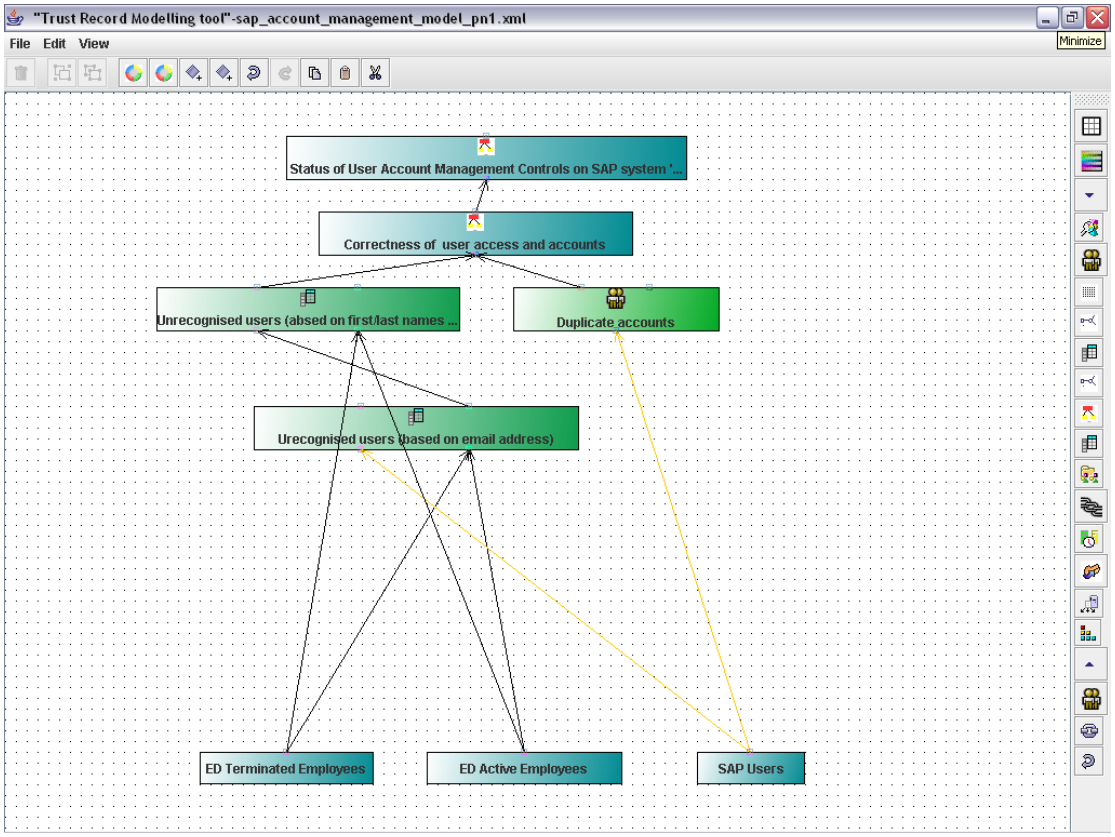


Figure 9 Model to check correctness of user accounts

In the past couple of years the audit team have also went through an exercise of identifying key control indicators that contribute to the early prediction of the effectiveness of certain application controls. These indicators have been implemented specifically for SAP systems, and have been used to gain an early visibility into the state of controls across different SAP systems. Some of these indicators have been replicated into our model, mostly measuring inactive users and locked accounts as a leading indicator of security effectiveness. The model with those measures is shown in figure 10.

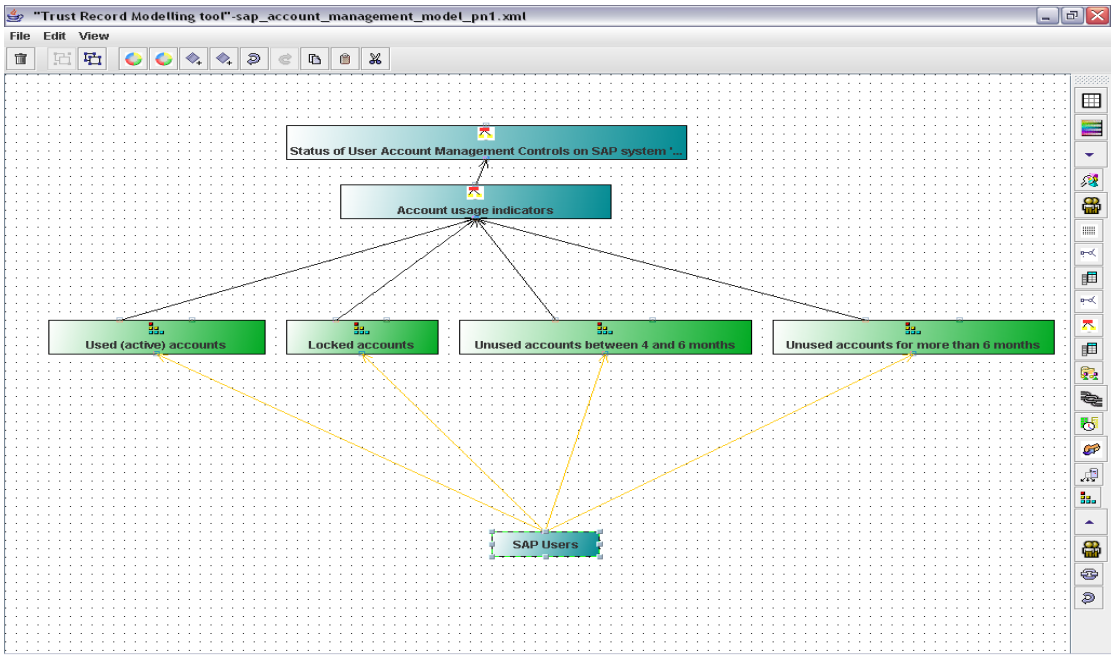


Figure 10 Leading account usage indicators

4.4 Segregation of Duty Analysis[5]

Segregation of duties is an internal control intended to prevent or decrease the occurrence of innocent errors or intentional fraud. This is done by ensuring that no single individual has control over specific combination of access in a business transaction. In general, the approval function, the accounting/reconciling function, and the asset custody function should be separated among employees. When these functions cannot be separated, a detailed supervisory review of related activities is required as a compensating control activity.

Before starting the testing of segregation of duty enforcement on the application the very first step is to identify if there exists a segregation of duty matrix that specifies what business functions are conflicting and cannot be assigned to the same individual. An example of such a matrix is shown in figure 11. This matrix should also include the mapping of these business functions to specific authorizations or roles that can be found in the application. Only if these two references exist the application analysis can be performed to identify if the authorisations are correctly assigned to the application users.

[5] This particular part of the model based analysis was not used during the pilot as the testing of this control was not in the scope of the audit engagements that the approach was being tested. We are planning to concentrate on this in the future.

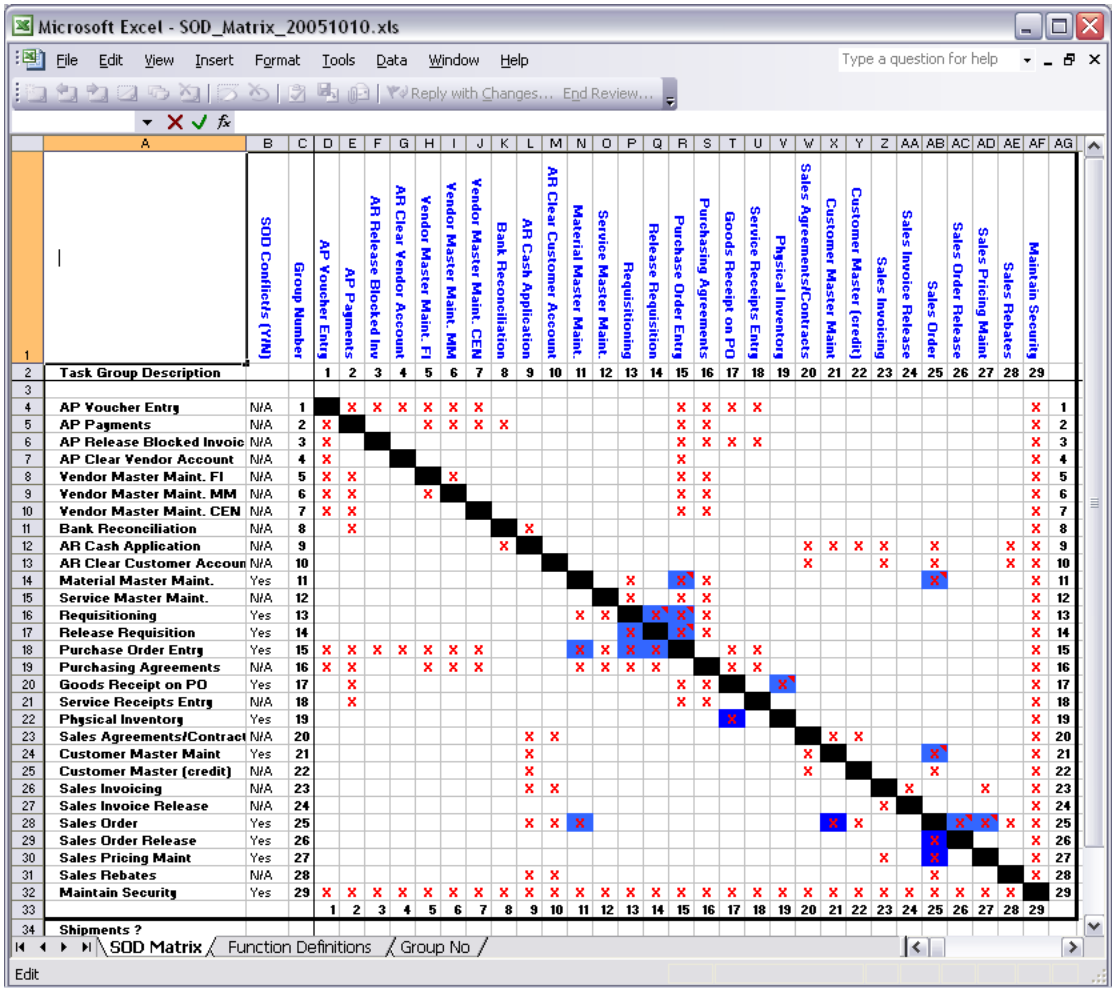


Figure 11 An example of segregation of duty matrix

When creating the analysis model below in figure 12 we are assuming that such references exist in a form that can be easily transferred to our centralized database. From the application side we need additional information about the authorisations/roles that are assigned to an individual user.

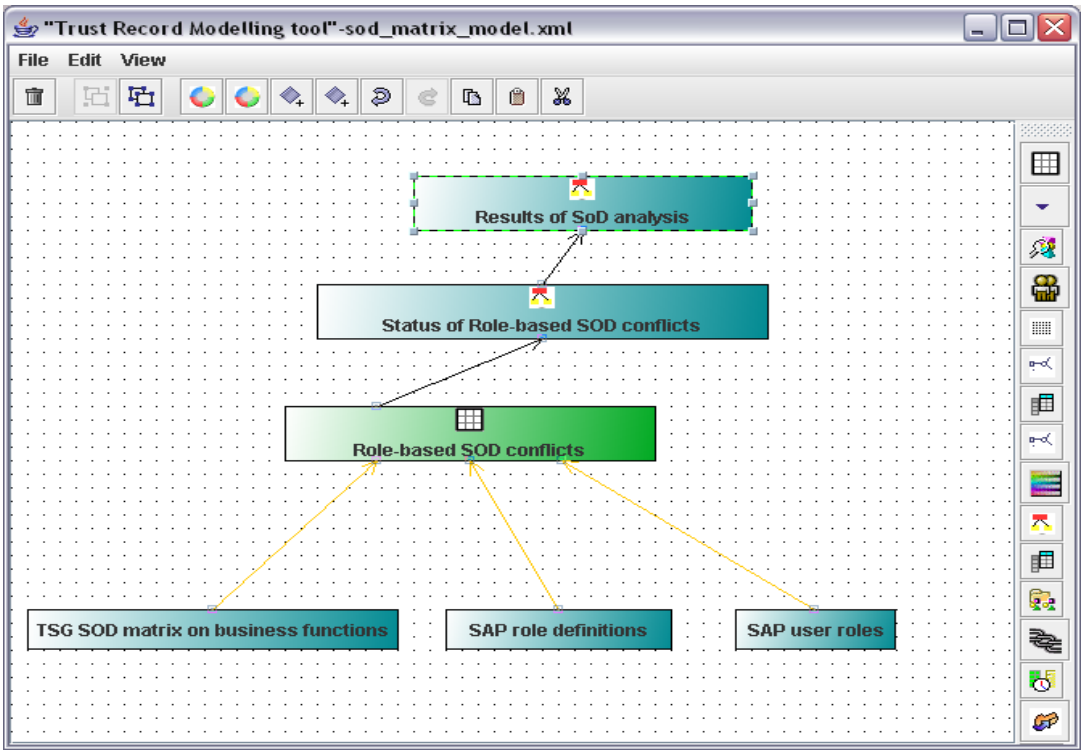


Figure 12 Segregation of duty analysis model

Based on the three input data sources the segregation of duty analysis model then includes one analysis node that is specifically designed to find conflicts in authorisation assignments corresponding to the ones defined in segregation of duty matrix.

The result of this analysis below in figures 13 and 14 shows the conflicts found including the individual account that has those conflicts, the actual conflicting authorisations, and the number of conflicts overall and per individual user.

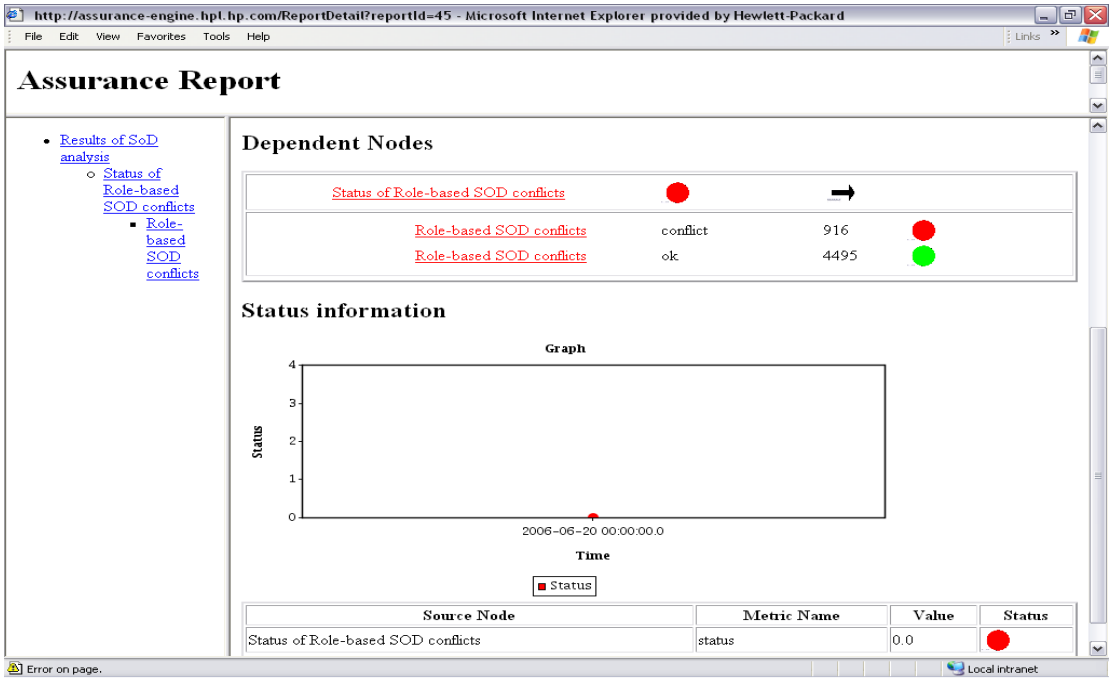


Figure 13 Top level results of the segregation of duty analysis

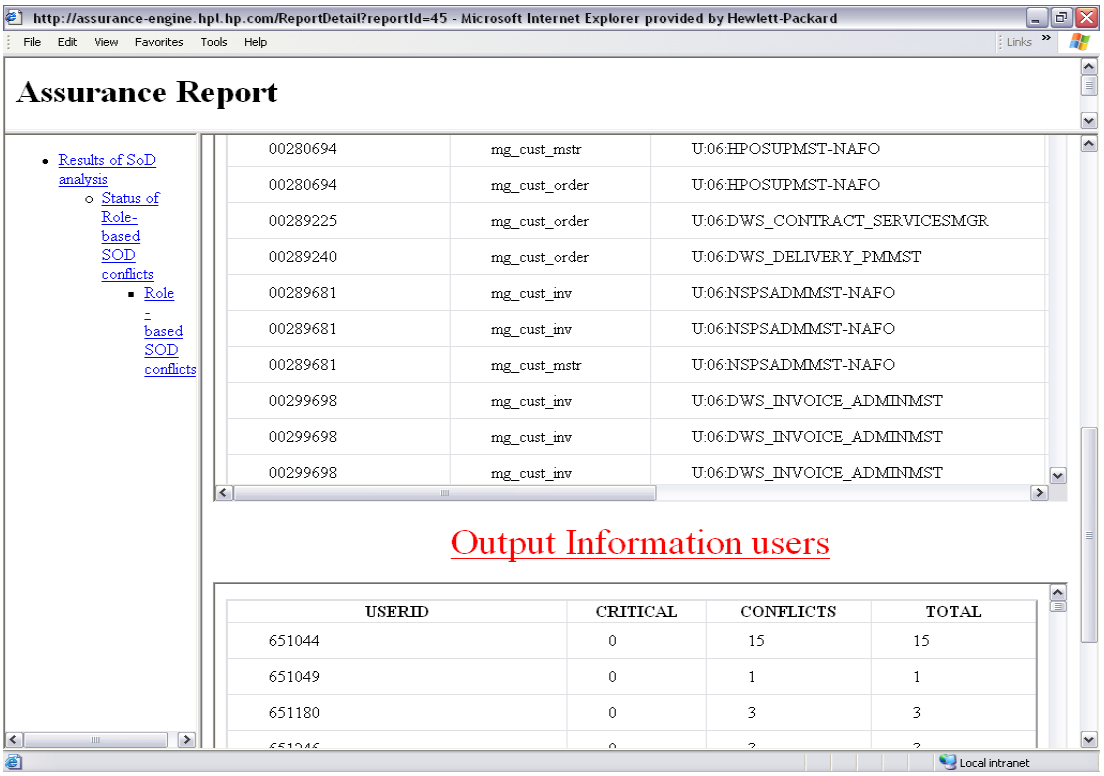


Figure 14 Detailed results of the segregation of duty analysis

4.5 Overall Model and Benefits

Figure 15 shows an overall account management model that includes all the separate models described in the five control areas. The full model as well as parts of the overall model was deployed to analyse the evidence of control effectiveness during 4 HP SOX application audits. Both auditors and the application owners have recognized a number of benefits of an automated model-based approach. During the four audit engagements the technology already helped save time and allowed auditors to concentrate more on problematic controls including out of scope controls. There are still some minor issues to fix usually related to the uniformity of data but the approach has been recognised as a great step in the right direction:

- for auditors this approach saves a lot of time and effort because it minimizes the amount of manual tests they have to do; in some cases more than 50%[6];
- because the results are available continuously (or the analysis can be performed just before an audit), it allows for auditors to see where the problems are and concentrate their efforts on problematic areas only;
- for application owners the analysis based on the model gives continuous view of how controls are working in regards to the set security policies or SOX control requirements; it also saves time required from application owners to prepare for audit since all data is already collected to support the analysis.

[6] For the account termination control around 80% of time saved; for the approval process around 50% of time saved when the application matched the model requirements; because of the constraints of time in each of the audit engagements the SOD analysis was not evaluated.

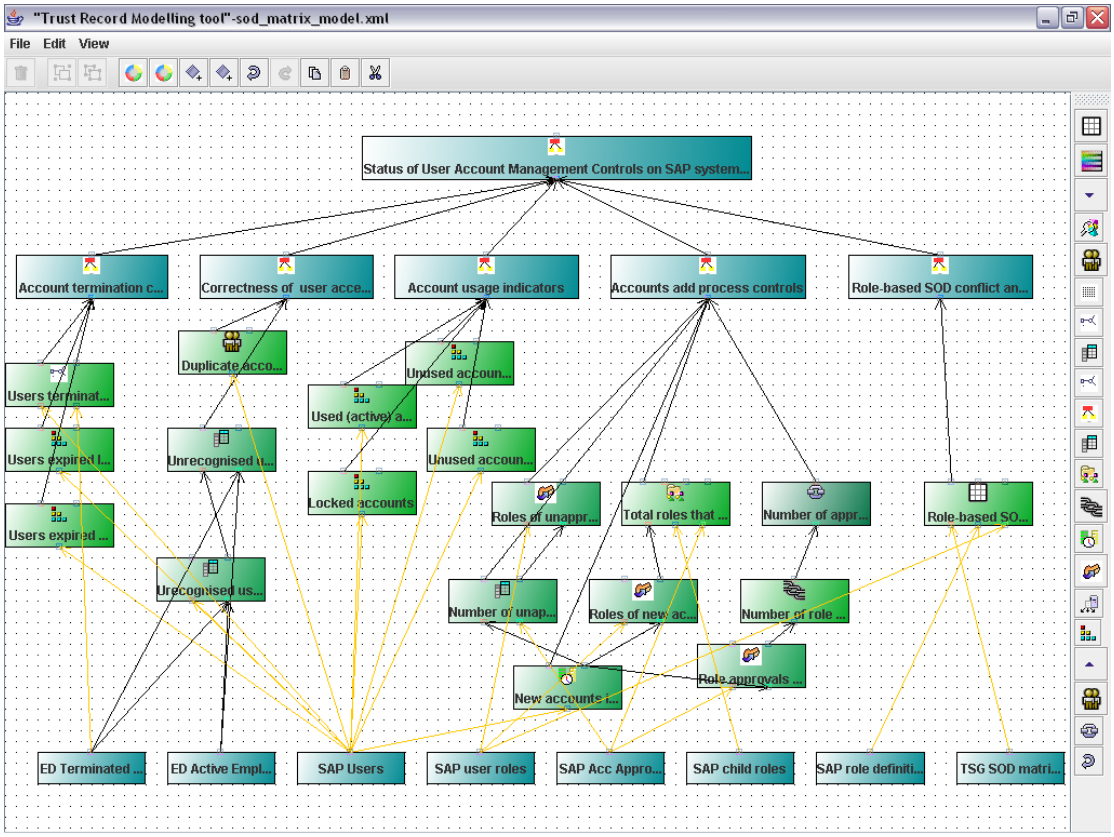


Figure 15 Overall model for analysis of account management controls

5 Other Control Domains

As we described previously IT audits are usually divided into three separate sets: application audits, database audits and infrastructure/data centre audits. In the previous sections we have described the models and the types of analysis that was used to continuously monitor controls during application audit pilot. This provides view of how individual or a set of applications are managed and what the exposed risks are. For a business owner or system manager to get a full picture of the overall IT environment it is necessary to also understand what the risks are the level below the applications such as databases and operating systems, networks and so on. Auditing of the controls at this level often falls within the scope of infrastructure auditors.

Based on the auditors' control matrix for the infrastructure we have started work on creating the corresponding models separately for database control analysis and for the lower level infrastructure controls such as the ones for server, operating systems and data centre management. We are currently working with HP IT service owners and internal infrastructure auditors to further enhance the models and also to understand the data sources required. We have been concentrating on security controls mainly, and have applied experimental models on Oracle databases and Linux/Windows servers to analyse mostly account management controls and gather metrics of account usage and password settings.

As the previous sections have demonstrated the model based analysis allows for high level of flexibility in terms of the different types of analysis that can be included in the model and also in terms of variety of data sources that the analysis can be applied to. In the application audit pilots we have concentrated mostly on security controls, mainly because of the expertise of our team and because we found the data sources were more easily available to analyse those types of controls. Besides security controls the set of internal controls audited by application and infrastructure auditors also include maintenance, integrity, and availability. Although we have not yet tried to capture these controls in our models, we believe that some aspects of the testing and analysis for these controls can be automated using our approach.

6 Conclusions

Compliance and heightened demands for improved corporate governance and fiscal transparency are not one-time events. Companies are increasingly calling on internal auditing to help improve performance by identifying areas of operational inefficiencies, risks in outsourcing environments, and fraud. The only way internal auditing can meet these demands — without growing its audit department significantly — is through the effective use of technology. In this document we presented how model based assurance technology has been used within HP IT audits to model the internal controls and automate analysis on the effectiveness of the implemented controls in real IT environments.

The results of such analysis are useful not just to the auditors but also to the system/application owners. Continuous analysis of controls based on the created models provide for timely, sometimes immediate, identification of anomalies or control gaps, and, once these gaps are identified, actions can be taken by the system owners to identify and correct problems before they get out of control. Being used over longer periods continuous model-based control analysis can help validate the adequacy of the mitigating controls, and so achieve greater effectiveness of the controls and ultimately better management of the risks.

References

[1] Sarbanes Oxley Act, http://www.sarbanes-oxley.com

[2] Staff Questions and Answers, Auditing Internal Control over Financial Reporting, Public Company Accounting Oversight Board - PCAOB, May 16, 2005, Q50, page 17, http://www.pcaobus.org/Standards/Staff_Questions_and_Answers/Auditing_Internal_Control_over_Financial_Reporting_2004-05-16.pdf

[3] A Baldwin, Y Beres, and Simon Shiu. Trust Record: Model based assurance, HP Labs Technical Report, 2005. http://lib.hpl.hp.com/techpubs/2005/HPL-2005-145.html

[4] A Baldwin, Y Beres, and Simon Shiu. Trust Record: High Level Assurance and Compliance. In proceedings of the 3rd International Conference on Trust Management, LNCS 3477, Paris, May 2005

[5] COSO: The Committee of Sponsoring Organisations of the Treadway Commission, http://www.coso.org

[6] HP, The HP IT Service Management (ITSM) Reference Model, 2003

[7] ITGI, Control Objectives for Information and Related Technologies (COBIT), 3rd edition, 1998

[8] ACL Audit Analytics Technology, the technical paper by ACL Services Ltd. http://www.acl.com/pdfs/ACL_Technology.pdf

[9] Payment Card Industry Data Security Standard

http://usa.visa.com/download/business/accepting_visa/ops_risk_management/cisp_PCI_Data_Security_Standard.pdf

[10] Health Insurance Portability and Accountability Act of 1996. http://www.legalarchiver.org/hipaa.htm

[11] A Baldwin, Y Beres, and Simon Shiu. Using assurance models to aid the risk and governance lifecycle. Submitted to BT Technical Journal.