Proactive Fraud Detection in Enterprise Systems

Kishore Singh, Peter Best, Joseph M. Mula
School of Accounting, Economics and Finance, USQ, Australia

Abstract

Fraud is a multi-billion dollar industry that continues to grow annually. Many organisations are poorly prepared to prevent and detect fraud. Fraud detection strategies are intended to quickly and efficiently identify fraudulent activities that have circumvented preventative measures so that an organisation can take appropriate corrective action. This paper investigates the use of enterprise system audit trails to support fraud detection. A framework for analysing these audit trails for potential fraud is developed. Key components of the framework include; defining the data requirements for fraud detection; and creating a catalogue of fraud symptoms. SAP, a market leading enterprise system is examined. We propose a MCL-based approach for proactive fraud detection that entails periodic extraction and analysis of SAP accounting audit trails for potential fraud. The paper emphasises the importance of visual presentation of information as it is this characteristic that enables an auditor to effectively and efficiently find trends, correlations, and identify patterns of activity that may lead to important conclusions regarding potentially fraudulent activities.

Keywords: fraud detection, enterprise system, continuous monitoring, audit trail

1. Introduction

Fraud is inherent in all organisations. Edwin H. Sutherland, a criminologist at Indiana University, coined the phrase *"white-collar crime"* in 1939 (Wells, 2008). Donald R. Cressy, a student of Sutherland, was especially interested in embezzlers, whom he referred to as "trust violators". He was intrigued by what led these people to be overcome by temptation. Upon completion of his work, he developed the classic model for the occupational offender. This model or hypothesis, namely the "fraud triangle", underpins the theoretical foundation for this study.

Fraud within organisations is a multi-billion dollar industry. Consequently, it is of major concern to industry and government (Best, 2005; Coenen, 2008). Fraud costs the Australian economy approximately $3 billion annually, and its frequency and financial impact continues to grow (KPMG, 2008; Standards Australia, 2008). Many organisations are poorly prepared to prevent and detect fraud (KPMG, 2004, 2007, 2008, 2009). Fraud prevention is not infallible, therefore fraud detection is crucial. Fraud detection strategies are intended to quickly and efficiently identify those frauds that have circumvented preventative measures so that an organisation can take appropriate corrective action (Standards Australia, 2008).

A review of various fraud surveys revealed that fraud is a crisis that is being faced by organisations internationally. Of all frauds detected in organisations, only 17% were attributed to the internal audit function (PwC, 2009). Whilst internal audit was the primary method of detecting fraud, the trend was that fewer frauds are being consistently detected. Opportunities to commit fraud are increasing, yet insufficient resources are being deployed to improving internal controls. Many organisations are considering the use of information technology (IT) to detect fraud (KPMG, 2008). Using IT to proactively detect fraud enables organisations to monitor and analyse large transaction datasets in real or near real time (Alles, et al., 2006), a task that cannot practically be accomplished by an internal auditor. A study of the literature reveals that there is a need for further research into proactive fraud detection that uses technology to rapidly analyse large sets of transaction data (Debreceny, et al., 2005).

2. Conceptual Fraud Model

Donald R. Cressy studied the circumstances that led employees to be so overcome by temptation that they were driven to violate their position of trust (Coenen, 2008; Wells, 2008). His work provided valuable insight into why people commit fraud and it led to the development of the fraud triangle. The three key elements of the fraud triangle are; pressure (usually an un-shareable need), rationalisation (of personal ethics), and opportunity (and knowledge to commit the fraud).

While all three elements of the fraud triangle must be present in order for a fraud to be perpetrated, the concept of opportunity is the main factor that provides a basis for this study. Opportunity and its antecedent characteristics are identifiable in an enterprise system. These characteristics can therefore be used to proactively detect fraud by analysis of an enterprise systems transaction data and audit trails.

Presuming that the elements of pressure and rationalisation pre-exist, a fraudster will actively seek opportunities to steal an asset (Wells, 2002; Wells, 2003). The conceptual model for this study defines the fundamental nature of fraud; and its detection (Figure 1). Firstly, the model identifies factors that motivate an individual to perpetrate fraud. Fraud triangle theory states that three key elements need to be present for a fraud to occur i.e. pressure, rationalisation and opportunity. The model describes the mental activity that fraudsters experience prior to perpetrating a fraud. They may mentally enact several fraud scenarios until a suitable one is found. Once a fraudster determines "what to steal" i.e. services, goods or cash, the next decision is "how to steal it" (Figure 2). A fraudster has to determine a specific method of perpetrating fraud. The chosen method may entail a series of steps taken to achieve the desired outcome of perpetrating a fraud; and concealing it to avoid detection.

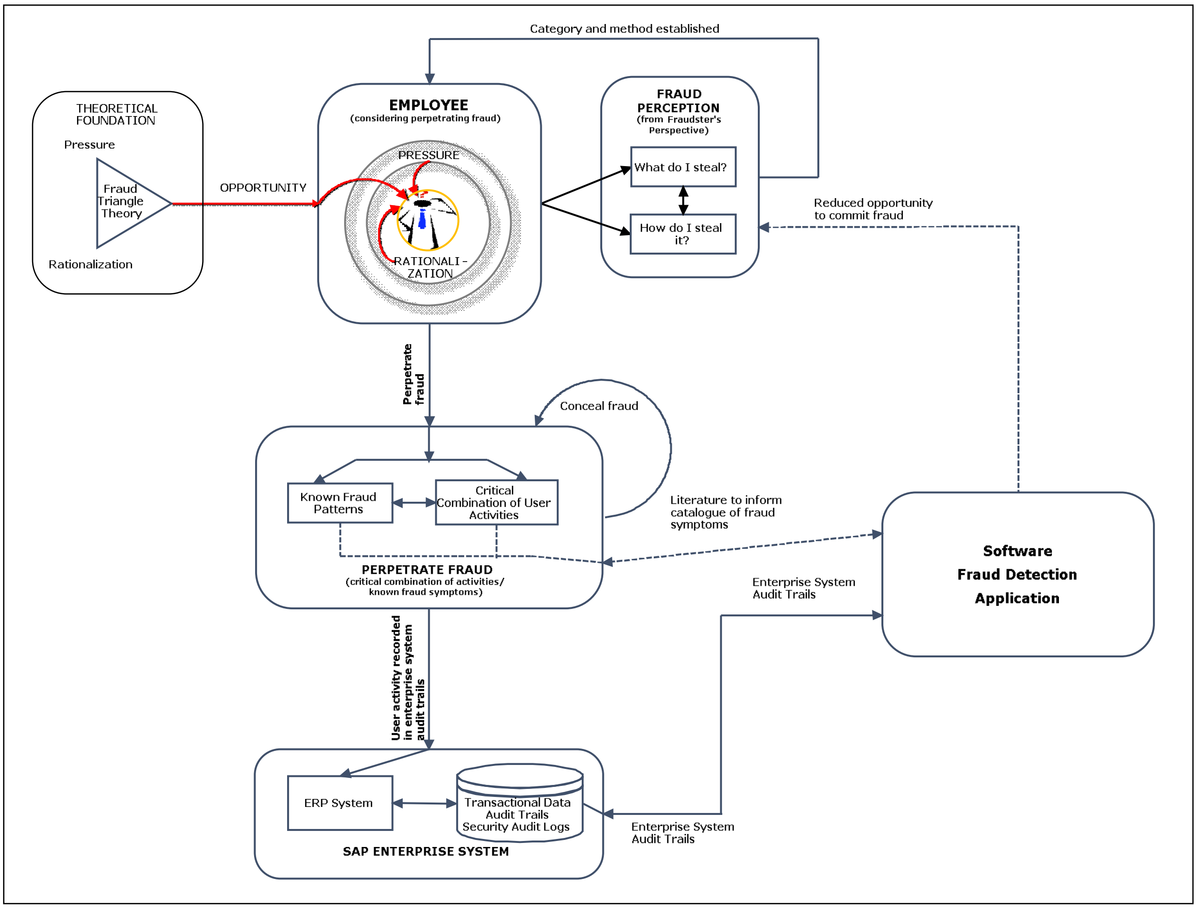
Category and method established
THEORETICAL FOUNDATION
Pressure
Fraud Triangle Theory
Rationalization
OPPORTUNITY
EMPLOYEE
(considering perpetrating fraud)
PRESSURE
RATIONALI-ZATION
FRAUD PERCEPTION
(from Fraudster's Perspective)
What do I steal?
How do I steal it?
Reduced opportunity to commit fraud
Perpetrate fraud
Conceal fraud
Known Fraud Patterns
Critical Combination of User Activities
PERPETRATE FRAUD
(critical combination of activities/ known fraud symptoms)
Literature to inform catalogue of fraud symptoms
Software Fraud Detection Application
Enterprise System Audit Trails
User activity recorded in enterprise system audit trails
ERP System
Transactional Data Audit Trails Security Audit Logs
SAP ENTERPRISE SYSTEM
Enterprise System Audit Trails

Figure 1: Conceptual fraud model

Figure 2 describes how fraudsters perceive fraud. An example of an accounts payable (AP) fraud is the theft of money that an organisation intends to pay to a vendor namely **Employee/What do I steal/Money/How do I steal money/divert payments made to vendors**. Another example is the theft of goods on order by an organisation namely **Employee/What do I steal/Goods/How do I steal goods/Goods on order from vendors (divert to personal address)**.

Secondly, the model focuses on the detection of fraud in an organisation. This is achieved by identifying types of frauds that can occur; creating a catalogue the fraud symptoms; and using computer software to detect fraud symptoms.

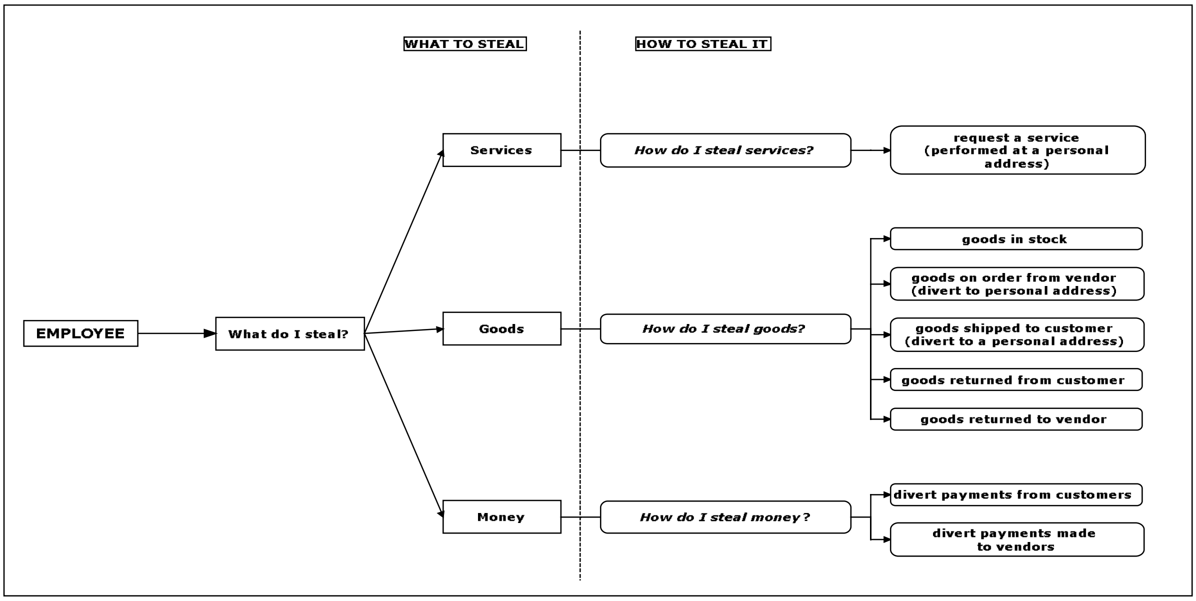
WHAT TO STEAL
HOW TO STEAL IT
EMPLOYEE
What do I steal?
Services
Goods
Money
How do I steal services?
How do I steal goods?
How do I steal money?
request a service (performed at a personal address)
goods in stock
goods on order from vendor (divert to personal address)
goods shipped to customer (divert to a personal address)
goods returned from customer
goods returned to vendor
divert payments from customers
divert payments made to vendors

Figure 2: Fraud perception model (FPM)

The conceptual model provides an understanding of the nature of fraud symptoms and its detection in enterprise systems. The next sections examine proactive fraud detection and continuous monitoring strategies that potentially support a reduction in the incidence of fraud in organisations.

3. Proactive Fraud Detection

Proactive fraud detection requires continuous monitoring of an organisations transaction data. Continuous monitoring increases the probability of detecting fraudulent activities (Coderre & Warner, 1999; Potla, 2003). The traditional or manual audit approach is limited because it reviews only a small percentage of a large population of transactions. Large accounting data files with several thousands of transactions are difficult to analyse or monitor manually in real-time. The alternative therefore is to automate this process by using information technology (Broady & Roland, 2008).

Every organisation must incorporate consideration of fraud risks within their risk management processes. Common fraud schemes, preventive measures and their symptoms or patterns are adequately documented in the literature (ACFE, 2010; Albrecht, et al., 2009; Coenen, 2008; Wells, 2008). Several vendor fraud schemes have been identified in the literature. For example an employee may create a fake vendor in the system and submit false invoices for payment. The enterprise system may pay these invoices electronically directly into an employee's bank account (Best, et al., 2009).

Segregating vendor maintenance, invoice entry and payment can significantly reduce the risk of accounts payable frauds (Little & Best, 2003; Srinidhi, 1994). Poor, incomplete or a lack of segregation of duties can, however, often provide opportunities for fraud schemes (ACFE, 2010;

KPMG, 2008, 2009). Early detection of fraud can limit losses and prevent the recurrence of such activities. The Sarbanes-Oxley Act (SOX) has significantly increased corporate organisations responsibility for prevention and detection of financial fraud (Best, et al., 2009; ITGI, 2006), therefore executives are searching for improved ways to detect fraud (Tackett, 2007) by proactively using information technology. The essential steps in detecting fraudulent activities are understanding the business or operations; performing a risk analysis to identify the types of frauds that can occur; cataloguing the symptoms that the most likely frauds would generate; using computer technology to identify fraud symptoms; analysing the results; and investigating suspect transactions (Albrecht, et al., 2009).

Automated systems that continuously monitor for key fraud symptoms can be a major deterrent of fraud (Best, et al., 2009; Coderre & Warner, 1999; Potla, 2003). By analysing data and searching for specific patterns or combination of activities, fraudulent activities can be identified shortly after they occur. Data analysis techniques can be used to detect fraudulent activities that have already occurred as well as to proactively determine the propensity for frauds occurring in the future (Edge & Falcone Sampaio, 2009). Presently only 2.6% of organisations are using data monitoring to proactively detect fraud (Figure 3) (ACFE, 2010).

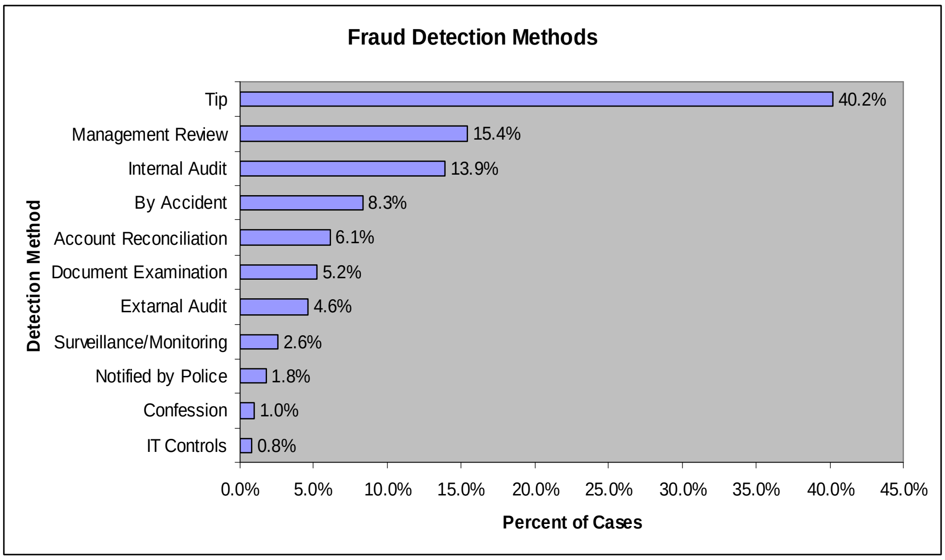

Figure 3: Detection of occupational fraud
Source: ACFE (2010)

4. Continuous Monitoring Strategies

Examining the transaction process from transaction entry through to posting in the general ledger has traditionally been conducted manually, or with Computer Assisted Audit Techniques (CAATs) on a retrospective and cyclical basis, usually many months after business activities have transpired (Flowerday & von Solms, 2005). This is a massive task as it may involve several thousands of transactions hence only a sample of transactions are examined. Even when using CAATs, the transactions are examined in batches and generally only on a sample. If transactions are examined in their entirety then it is usually done retrospectively.

Continuous monitoring is a way to provide constant monitoring of transaction data in a real or near real-time basis against a set of predetermined rule sets (Kuhn Jr & Sutton, 2010). It enables auditors to provide a degree of assurance on information shortly after disclosure (Rezaee, et al., 2002). It is a step in the path of the evolution of the financial audit from manual to computer-based methods. Widespread adoption of computer-based accounting information systems in general, and Enterprise Resource Planning (ERP) systems in particular, has contributed to the increasing demand for continuous monitoring (Vasarhelyi, et al., 2004). Two major approaches to continuous monitoring and auditing exist. These are Embedded Audit Modules (EAMs), and Monitoring and Control Layer (MCL).

Embedded Audit Modules (EAM)

EAMs are software modules that are built into application programs and are specifically designed to continuously capture and monitor audit related information (Groomer & Murthy, 1989). If a pre-programmed constraint is violated an alert is generated, an auditor is informed, and transaction data is saved in a file (Best, et al., 2009; Debreceny, et al., 2005; Groomer & Murthy, 1989; Weber, 1999)

Weber (1999) describes EAMs as modules that are placed at specific points within a system to gather material information about events or transactions. EAMs are therefore intended to detect and capture data as transactions are processed in the enterprise system. When a violation occurs the offending transaction can either be rejected or allowed and an error is logged. ERP systems are designed to process transactions efficiently and promptly. It is therefore not practical to disallow every offending transaction from being processed. Depending on the severity of the violation, some transactions could be conditionally processed whilst others are rejected. The level of severity of errors that would cause a transaction to be rejected needs to be negotiated and accepted by the client organisation (Groomer & Murthy, 1989).

Research seems to indicate that the EAM approach runs into several difficulties (Debreceny, et al., 2005; Kuhn & Sutton, 2006). Vasarhelyi and Halper (1991) expressed several challenges to their development and implementation, including issues related to design and utilization of system resources. Since EAMs are software applications they require computer processing time to execute. This imposes an overhead on the system which in turn negatively impacts the monitoring processes. Although this overhead can be overcome by adding additional hardware and software resources, these additional investments have costs associated with them. There is also the concern about having "foreign" software embedded within an organisations enterprise system, and this software being the responsibility of a third party (Alles, et al., 2006; Best, et al., 2009; Debreceny, et al., 2005). The maintenance of EAMs can also be difficult given the changes, updates and modifications that routinely take place in enterprise systems. There are also legal liability issues should the EAM damage the host enterprise system in some way, a liability that external auditors may be keen to avoid. These factors have impeded the adoption of EAMs in ERP systems (Debreceny et al. 2005; Alles et al. 2006).

Monitoring and Control Layer (MCL)

The Monitoring and Control Layer (MCL) introduced by Vasarhelyi et al. (2004) is an alternative continuous monitoring and auditing approach to EAMs. MCLs do not replace EAMs, instead they offer an alternative solution to cater for different circumstances (Kuhn Jr & Sutton, 2010). In this approach the continuous monitoring and auditing system is separate from the client's enterprise system. MCLs are stand-alone systems that rely on comparisons of extracted transaction data with pre-determined constraints that allow for continuous monitoring of systems and identification of violations (Du & Roohani, 2007).

The MCL primarily operates as a discrepancy-based audit monitoring tool i.e. audit by exception (Vasarhelyi, et al., 2004). The MCL continuously captures enterprise data and analyses it to detect any deviations from the norm. Whenever a significant exception is detected, an alarm is generated and sent to pre-determined compliance personnel by using relevant delivery technologies such as emails, telephone calls or pagers. When an alarm is delivered, compliance personnel will need to review the evidence in order to identify the underlying problem. Any further investigations are at the discretion of internal auditors.

The continuous monitoring system that makes up the MCL (i.e. workstations, operating systems, database and application software) resides outside the client's network and is controlled by the auditor. The system receives periodic data updates from the client's enterprise system, (i.e. not in real-time), that is processed inside the application. The system monitors key operational analytics, compares them with pre-defined standards and creates exception reports for any potential problems. Any violations that trigger automatic alerts to the auditor are stored inside the application and not inside the client's enterprise system.

MCLs are external systems that operate independently of the information system to be monitored but are linked into the system. They rely on comparisons of extracted transaction data with pre-determined constraints to identify violations. This separate design has profound implications for the design of a general model for continuous monitoring and auditing as it eliminates any conflict between the MCL and the enterprise system. The MCL approach is therefore a major facilitator for implementing continuous monitoring and auditing in enterprise systems.

Continuous monitoring systems capture and analyse enterprise system audit trail data to detect any deviations from the norm. Audit trails are a chronological record of activities performed in an enterprise system and are therefore an essential component in monitoring activities performed in an enterprise system. The usefulness of audit trails in detecting potentially fraudulent activities is examined next.

5. Audit Trails

Audit trails are records of users' activities within an information system (Best, 2005; NIST, 2005). Audit trails are maintained by the operating system and applications such as database systems and enterprise systems (Best, et al., 2004). The information captured in an audit trail is dependent on what events are being audited by the system (SAP-AG, 2009). In conjunction with appropriate tools and procedures, audit trails can assist in detecting potentially fraudulent activities.

Audit trails attempt to establish a chronological list of steps that are necessary to start a transaction through to its completion. Audit trails can range from being very simplistic to extremely complicated. The complexity depends on the number of steps involved in the transaction. For example, an audit trail on a payment of a vendor invoice begins with the receipt of the invoice. The invoice is tracked through accounts payable, all the way through to payment in order to settle the debt (Tatum, 2010)

Denning (1987) introduced the concept of using audit trails to detect anomalous user behaviour. Denning's model is rule-based and exploits audit trails to search for and report abnormal user behaviour. The basic objective of the model is to monitor audit records looking for deviations in usage. Audit trails may be reviewed: i) periodically, ii) as needed (triggered by a security event), iii) automatically in real-time, or iv) some combination of these. Audit trails can be used to retrospectively determined review what events occurred. Reviewers need to know what to look for i.e. what is normal activity and what is suspicious activity. Audit trail review is made easier if the audit trail can be analysed by user ID, terminal ID, application name, date and time, or some other set of parameters to run reports of selected information (NIST, 2005).

Audit trails provide an auditor a means to review user activities by allowing examination of the history of access by individual users or groups of users, showing actions performed or attempted. Analysis of audit trails may reveal activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous (Best, 2005; Best, et al., 2009; NIST, 2005). Audit trails can therefore be an effective tool in managing financial resources of an organization. Prior to analysing audit trail data, a framework needs to be developed to enable detection of potentially fraudulent activities.

6. Framework for Detecting Fraud

Perpetration of an accounts payable (AP) fraud requires the creation of a shell company and the submission of fictitious invoices to an organisation for payment (Best, et al., 2009; O'Gara, 2004; Wells, 2002a). To successfully perpetrate this type of fraud the fraudster needs to access to the following enterprise system elements; i) creation or modification of vendor master records; and ii) invoice entry sub-system (Best, et al., 2009; Narayan, 2008; Padhi, 2010).

Vendor master records can be created or modified in the following ways; i) create a fake vendor ; ii) temporarily modify an existing vendor (flipping); iii) permanently modify an existing vendor; or iv) use a one-time account (Best, 2008; O'Gara, 2004; Singleton, et al., 2008).

Invoices can be entered in an enterprise system in the following ways; i) create a fake invoice; ii) use a legitimate invoice; or iii) create or use a duplicate invoice (Best, 2005; Singleton, et al., 2008).

Key components of the framework for proactive fraud detection include defining data requirements for fraud detection; and creating a catalogue of fraud symptoms. The catalogue of fraud symptoms comprises critical combinations of user activities and known fraud symptoms.

Critical Combinations

Many frauds occur because fraudsters exploit the lack of internal controls or they may override existing internal controls that are poorly implemented. For example, an employee that creates or modifies a vendor master record should not be able to enter an invoice. Having this capability does not indicate that a fraud has taken place, but it does create an opportunity for a fraud to be perpetrated. By detecting these critical combinations of user activities; i) an auditor can further investigate transactions that match known fraud symptoms, or appear otherwise anomalous and, ii) an organisation can take steps to correct the situation thereby reducing the probability of future fraud.

The concept of separating critical business activities in order to reduce fraud is termed segregation of duties. In its simplest form, the Segregation of Duties (SoDs) principle states that sensitive tasks should be divided into two or more steps with each step being performed by a different user (Li, et al., 2007). This study supports the following principles of SoDs within the accounts payable function as proposed by Little and Best (2003); i) users who can create and modify master records should not be able to post transactions; and ii) payments should be performed by someone other than the person who enters vendor invoices (Figure 4).

Known Fraud Symptoms

Accounts payable fraud schemes occur when a fraudster causes an organization to issue a payment by submitting invoices for fictitious goods or services, inflated invoices, or invoices for personal purchases. Activities that violate segregation of duties are indicators of potential fraud and require further investigation. These activities are investigated to determine whether they match known fraud symptoms, or appear otherwise anomalous. Methods to detect several known accounts payable fraud symptoms are specified in Table 1.

Enterprise systems software are available from several vendors, however SAP has consistently been the market leader for several years, accounting for 22.4% of market share (Lager & Tsai, 2008; SAP, 2010). Several Fortune 500 companies use SAP exclusively for their core day to day operations including accounting and financial applications, procurement, order processing and supplier management, inventory management and HR management and payroll functions (BOS, 2009; Gartner, 2010). Consequently, the focus of this paper is on fraud detection in the SAP enterprise system. In the next section we examine the level of support provided for fraud detection in SAP.

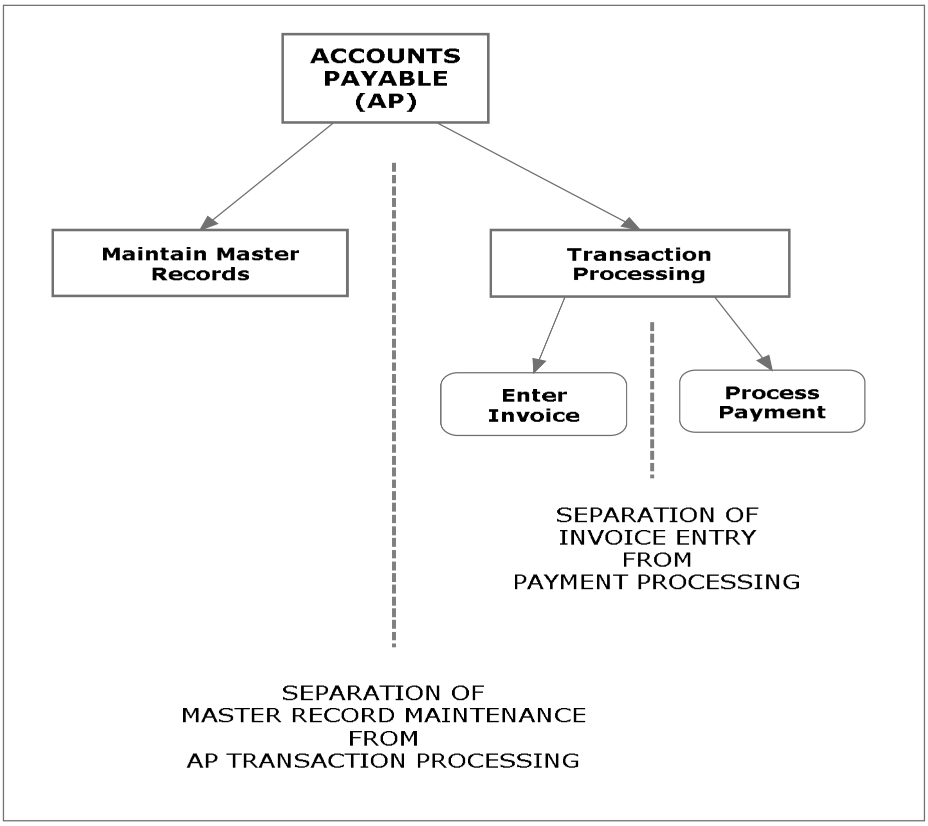
ACCOUNTS PAYABLE (AP)
Maintain Master Records
Transaction Processing
Enter Invoice
Process Payment
SEPARATION OF INVOICE ENTRY FROM PAYMENT PROCESSING
SEPARATION OF MASTER RECORD MAINTENANCE FROM AP TRANSACTION PROCESSING

Figure 4: Critical combinations of AP activities
Source: adapted from Little and Best (2003)

7. SAP Support for Fraud Detection

SAP audit trails provide detailed descriptions of functions performed within an enterprise system. Each function in SAP has a transaction code associated with it. A transaction code (or t-code) consists of letters, numbers, or both (for example, FB60 – Enter Vendor Invoice). A transaction code is a shortcut that takes the user directly to a SAP application rather than having to navigate through the menu system (Padhi, 2010). Each transaction code executed by a user is recorded in the audit trail (Best, 2000). The audit trail data required for this study is stored in several tables within the SAP enterprise system (Figure 5).

Changes to master records are stored in two tables, CDHDR Change Document Headers, and CDPOS Change Document Items (Best, 2005; Best, et al., 2009; Hirao, 2009; Padhi, 2010). Changes to master records include creation and deletion of master records and changes to fields. For every change document number, there is a corresponding change document item in the CDPOS table. Accounting audit trails are stored in tables BKPF – Accounting Document Header, BSEG – Accounting Document Line Item, SKAT – General Ledger Account Texts, and LFA1- Vendor General Data. Tables BKPF and BSEG store the posting history for both general ledger accounts and subsidiary ledger records, thereby facilitating both integration of data and automatic reconciliation of subsidiary ledgers with reconciliation accounts. General ledger account texts (names) are stored in table SKAT.

Vendor general data including vendor name, date created and creating user are stored in table LFA1. The relationships between the various SAP tables are shown in Figure 6. These relationships are exploited by the proactive fraud detection methodology developed in this study.

Table 1: Methods to detect known AP fraud symptoms

Symptom	General Detection Strategy
Change in vendor payment details followed by a change back to the original after a short time (flipping) and payments are made in the interim period	§ Detect changes to vendor master data that result in a vendor having different bank details over a period of time. Payments of invoices are made in the interim period. Previous bank details are subsequently reinstated after being updated with new details.
Duplicate transactions	§ Check if the same payment details are used by more than one vendor
Invoices with round dollar amounts	§ Extract all invoices with round dollar amounts (e.g. $1000.00)
Invoices with amounts consistently below approval limit	§ Extract all vendors with multiple invoices below approval limit (e.g. several $999 payments to vendor when limit is $1000)
Vendors with payments that exceed their 12-month average by a significant amount	§ Extract all vendors where payments exceed 12-month average by a percentage e.g. 200%
Vendors with payments exceeds the last largest payment by a significant amount	§ Extract all vendors where payment is larger than the last largest payment by a percentage e.g. 200%
Vendors with similar names	§ Extract all vendors whose names are similar to other companies
Vendors that become active after long periods of being dormant	§ Extract all vendors that become active after long periods of inactivity
Same vendor having different payment details	§ Extract all vendors with multiple master records, each having different payment details § Check for multiple payments using different bank account details
Multiple vendors sharing the same payment details	§ Extract all vendors that share the same payment details

Source: adapted from (Best, et al., 2009; Lanza, 2003; Wells, 2008)

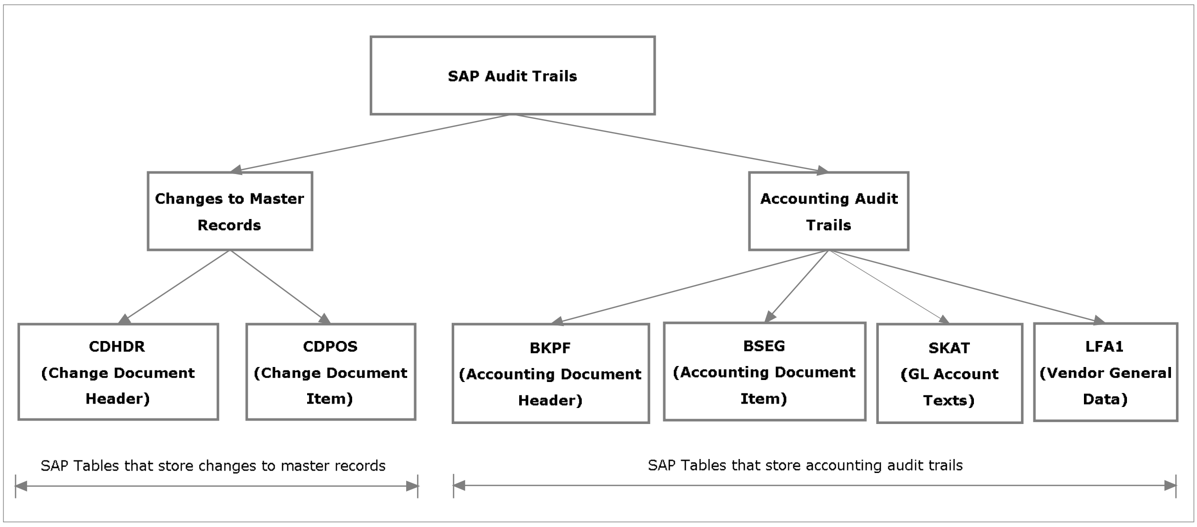
SAP Audit Trails
Changes to Master Records
Accounting Audit Trails
CDHDR (Change Document Header)
CDPOS (Change Document Item)
BKPF (Accounting Document Header)
BSEG (Accounting Document Item)
SKAT (GL Account Texts)
LFA1 (Vendor General Data)
SAP Tables that store changes to master records
SAP Tables that store accounting audit trails

Figure 5: SAP audit trails

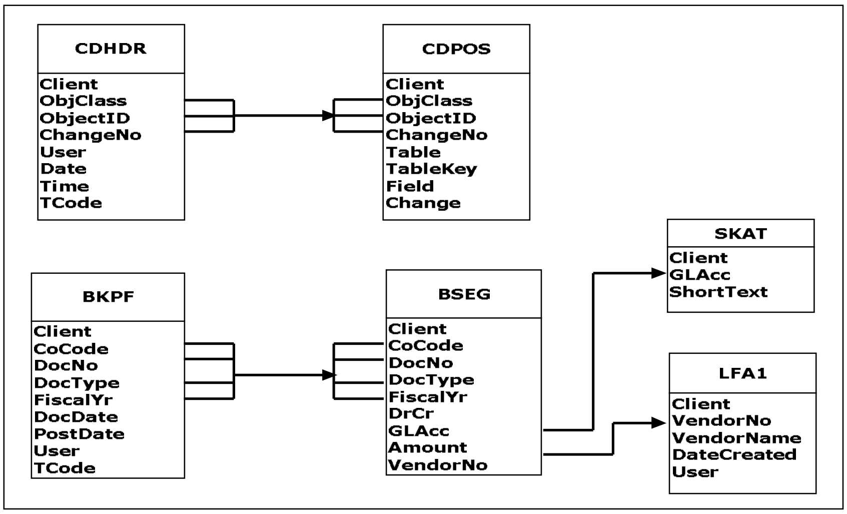

Figure 6: Relationship between SAP tables

8. Fraud Detection in SAP

The SoDs principles previously discussed can be detected in SAP by examining t-codes of functions performed by users. A list of t-codes pertinent to each of the two SoDs principles is listed in Table 2.

In order to detect a violation of the first SoDs principle it is necessary to identify users that perform vendor maintenance (FK01, FK02, XK01, XK02) and enter invoices (FB60, F-43, FB01, FB10) or post payments (F-53, F-58, F110). In order to detect a violation of the second SoDs principle it is necessary to identify users that enter invoices (FB60, F-43, FB01, FB10) and post payments (F-53, F-58, F110). User activities that violate these two SoDs principles may be further investigated to determine whether they match known fraud symptoms, or appear otherwise anomalous by extracting data from the previously mentioned SAP tables. Table 2 informs this process.

Table 2: SAP transaction codes

T-Code	SAP Description
Vendor Maintenance	
FK01	Create Vendor (Accounting)
FK02	Change Vendor (Accounting)
XK01	Create Vendor (Centrally)
XK02	Change Vendor (Centrally)
Enter Invoice	
FB60	Enter Vendor Invoice
F-43	Enter Vendor Invoice: Header Data
FB01	Post Document (allows posting of any financial transaction)
FB10	Invoice/Credit Memo Fast Entry
Post Payment	
F-53	Post Outgoing Payment
F-58	Post Payment with Printout
F110	Automatic Payments

Data describing user activities is well-documented in the audit trails of SAP enterprise systems. Analysing user activities for potential fraud, however, is a difficult task if done manually. Automated systems that continuously monitor for key fraud symptoms can be a major deterrent of fraud (Best, et al., 2009; Coderre & Warner, 1999; Potla, 2003). By analysing transaction data and searching for specific patterns or combination of activities, fraudulent activities can be identified shortly after they occur. Computer based data analysis techniques can be used to detect fraudulent activities that have already occurred, as well as proactively determining the propensity for frauds occurring in the future (Edge & Falcone Sampaio, 2009). An automated methodology for proactive fraud detection is proposed in the next section.

9. Proactive Fraud Detection Methodology

Modern integrated enterprise systems may record several thousands of transactions daily. A significant issue often raised in the literature regarding continuous fraud detection systems relates to information overload from alerts (Alles, et al., 2006; Alles, et al., 2008; Kuhn & Sutton, 2006). Therefore, simple detection of fraudulent activities is insufficient. It is imperative to develop innovative approaches for analysis and presentation of information to an auditor. The methodology developed for this study addresses these issues.

This study proposes a two phase MCL-based strategy for proactive fraud detection in an organisation's SAP enterprise system. In phase one, transaction data is periodically extracted from SAP. The requisite data is extracted from SAP tables CDHDR (change document headers), CDPOS (change document items), BKPF (accounting document headers), BSEG (accounting document items), SKAT (general ledger account texts), and LFA1 (vendor general data).

In phase two, extracted transaction data is analysed by a software application. The analysis consists of two stages. In stage one, critical combinations of user activities are identified. These activities violate SoDs and require further investigation. In stage two, user activities that violate SoDs are further investigated to determine whether they match known fraud symptoms, or appear otherwise anomalous. The volume of alerts produced by large transaction data sets may be difficult to interpret. Consequently, user profiling and 'drill-down' capabilities enhance useability by enabling an auditor to peruse detailed user or vendor centric activities.

Stage one involves routine analysis of transaction data. A summarised list of activities performed in the system provides the context of analysis for an auditor (Table 3 and Figure 7). Data visualization techniques are used to graphically represent this information in the form of charts, graphs, and link-node (network) maps. Representing large amounts of data visually enables an auditor to effectively find trends, correlations, and identify patterns of activity that may lead to important conclusions regarding potentially fraudulent activities.

Table 3: Activity summary

T-Code	Transaction Name	Activity
FB60	Enter Incoming Invoices	2305
F-53	Post Outgoing Payments	2135
FK02	Change Vendor (Accounting)	252

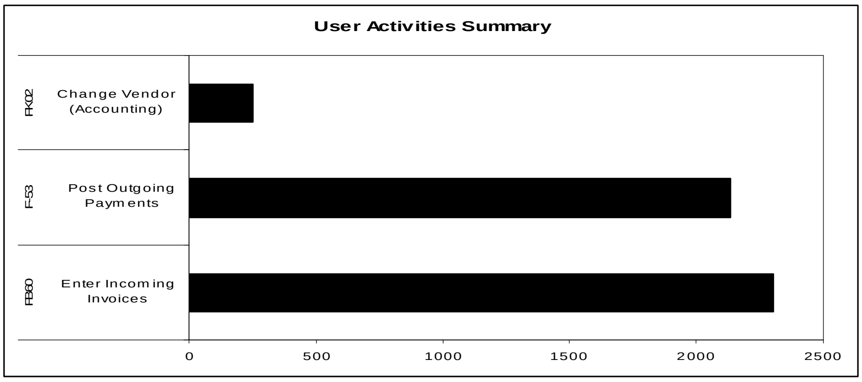

Figure 7: Activity summary

The next step in the analysis process is an investigation of user profiles. User profiles document combinations of transaction codes performed by individual users in the system (Table 4). A network (link-node) map visualises the relationships between users and the transaction codes they have performed. This may assist an auditor in visualising user activities and identification of users' violating SoDs (Figure 8).

Table 4: User profile

User	Transaction		
Watts	FB60	F-53	
Green			FK02
John	FB60		
Smith	FB60	F-53	FK02
Jones			FK02

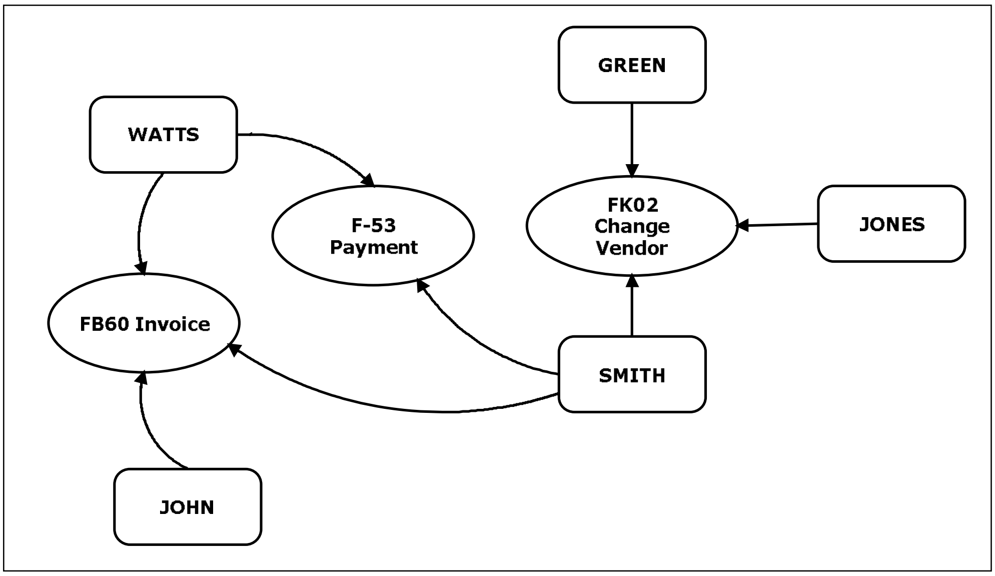

Figure 8: User profile

In stage two, user activities that violate SoDs may be further investigated by an auditor. An example of such an investigation entails detailed examination of activities performed by user SMITH, as this user has violated SoDs by performing vendor maintenance, invoice and payment activities (a critical combination). Drill-down facilities produce user specific reports and visualisations detailing activities performed by user SMITH (Figure 9). This process of combining visualisation with tabulated reports enhances an auditor’s ability to promptly identify potentially fraudulent activities without being overwhelmed with excessive information.

10. Limitations

Audit trails maintained in an SAP Enterprise System form the basis of the methodology developed in this paper. Consequently, the integrity of these audit trails is of vital importance in assessing the usefulness and accuracy of the fraud detection process. Furthermore, the detection process itself may generate incorrect results due to type I (false positive) or type II (false negative) errors. A false positive results in fraudulent transactions being classified as legitimate. A false negative results in legitimate transactions being classified as fraudulent.

System and security administrators that have the capability to create and/or maintain users may create fake users and act in their name. Users identified through proactive fraud detection must be investigated to determine whether they are real. Vendor frauds may also be perpetrated using duplicate vendor master records. In this situation flipping of a vendor's banking details is not required.

There is also the threat of collusion between users, such that no one user performs all required tasks to perpetrate a fraud. A combination of these methods may be used.

The proposed fraud detection methodology may not be useful in detecting fraud by system and security administrators, nor fraud perpetrated by two or more user in collusion. However, this methodology is intended to assist an auditor in early detection of fraudulent activities perpetrated by normal system users.

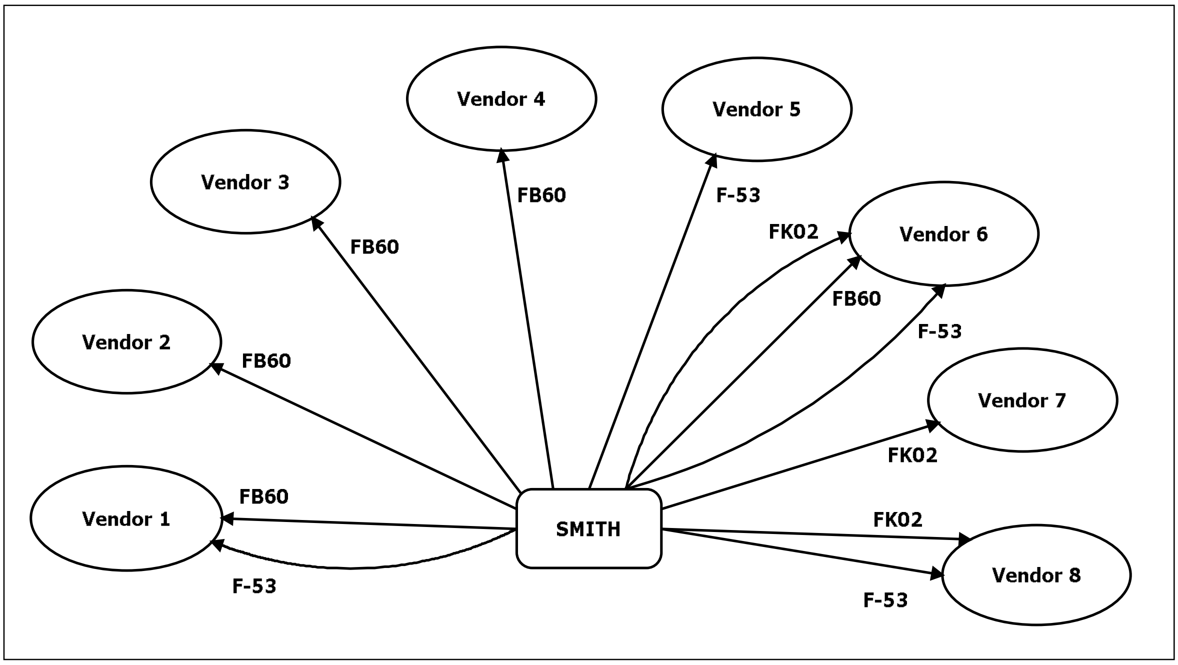

Figure 9: User-vendor activities

11. Conclusion and Future Research

This paper examines the feasibility of proactive fraud detection in enterprise systems, in general, and SAP in particular. The paper examines the level of support provided by SAP for fraud detection, and proposes a MCL-based methodology for proactive fraud detection. This paper presents significant advances over current Computer Assisted Audit Techniques (CAATs) by emphasizing the importance of visual presentation of information to an auditor. This is unlike current practice that involves analysis of lengthy reports or spreadsheets.

Although we have provided two visualisations that show; i) user profiles; and ii) user-vendor activities, future work will focus on drill-down capabilities that provide analysis of transactions performed by targeted users. Reporting will include fraud symptoms documented in Table 1. Visualisations will include graphs, charts and link-node diagrams. Future work will also focus on presenting a high-level view of activities performed in accounts payable (AP). The objective is to integrate multiple sources of information into a unified view that may assist an auditor to effectively and efficiently identify patterns of activity that may lead to important conclusions regarding potentially fraudulent activities.

References

ACFE (2010). Report to the Nation on Occupational Fraud and Abuse Retrieved 6/10/2010, from http://www.acfe.com/rttn

Albrecht, W. S., Albrecht, C. C., & Albrecht, C. D. (2009). *Fraud Examination* (3rd Ed. ed.): Thomson/South-Western.

Alles, M., Brennan, G., Kogan, A., & Vasarhelyi, M. A. (2006). Continuous monitoring of business process controls: A pilot implementation of a continuous auditing system at Siemens. *International Journal of Accounting Information Systems, 7*(2), 137-161.

Alles, M. G., Kogan, A., & Vasarhelyi, M. A. (2008). Putting Continuous Auditing Theory into Practice: Lessons from Two Pilot Implementations. *Journal of Information Systems, 22*(2), 195-214.

Best, P. J. (2000, 23-25 July 2000). *SAP R/3 Audit Trail Analysis.* Paper presented at the Sapphire 2000. 4th Annual SAP Asia Pacific Institute of Higher Learning Forum., Brisbane, Australia.

Best, P. J. (2005). *Audit Trail Analysis For Fraud Control With SAP R/3.* Paper presented at the Oceania Computer Audit, Control and Security Conference (CACS) 2005 Conference.

Best, P. J. (2008). SAP - Accounts Payable. On *ACC3101 - Accounting Information Systems*: USQ.

Best, P. J., Mohay, G., & Anderson, A. (2004). Machine-Independent Audit Trail Analysis – A Decision Support Tool for Continuous Audit Assurance. . *International Journal of Intelligent Systems in Accounting, Finance & Management 12*(2), 85-102.

Best, P. J., Rikhardson, P., & Toleman, M. (2009). Continuous Fraud Detection in Enterprise Systems through Audit Trail Analysis. *Journal of Digital Forensics, Security and Law, 4*(1).

BOS (2009). Benefits of Using SAP for Your Business Retrieved 08/11/2010, from http://www.bos.com.np/index.php?option=com_content&view=article&id=61:benefits-of-using-sap-for-your-business&catid=34:articles&Itemid=72

Broady, D. V., & Roland, H. A. (2008). SAP GRC For Dummies Available from http://library.books24x7.com.ezproxy.usq.edu.au/toc.asp?bkid=25161

Coderre, D., & Warner, P. D. (1999). Computer-Assisted Techniques for Fraud Detection. *CPA Journal, 69*(8), 57.

Coenen, T. (2008). Essentials of Corporate Fraud Retrieved from Books24x7 database Available from http://library.books24x7.com.ezproxy.usq.edu.au/book/id_24342/viewer.asp?bookid=24342&chunkid=0955093815

Debreceny, R. S., Gray, G. L., Jun-Jin Ng, J., Siow-Ping Lee, K., & Yau, W.-F. (2005). Embedded Audit Modules in Enterprise Resource Planning Systems: Implementation and Functionality. *Journal of Information Systems, 19*(2), 7-27.

Denning, D. E. (1987). An Intrusion-Detection Model. *Software Engineering, IEEE Transactions on, SE-13*(2), 222-232.

Du, H., & Roohani, S. (2007). Meeting Challenges and Expectations of Continuous Auditing in the Context of Independent Audits of Financial Statements. *International Journal of Auditing, 11*(2), 133-146.

Edge, M. E., & Falcone Sampaio, P. R. (2009). A survey of signature based methods for financial fraud detection. *Computers & Security, 28*(6), 381-394.

Flowerday, S., & von Solms, R. (2005). Continuous auditing: verifying information integrity and providing assurances for financial reports. *Computer Fraud & Security, 2005*(7), 12-16.

Gartner (2010). Gartner Says Worldwide Business Intelligence, Analytics and Performance Management Software Market Grew 4 Percent in 2009 Retrieved 27/10/2010, 2010, from http://www.gartner.com/it/page.jsp?id=1357514

Groomer, S. M., & Murthy, U. S. (1989). Continuous Auditing of Database Applications: An Embedded Audit Module Approach. *Journal of Information Systems, 3*(2), 53.

Hirao, J. (2009). *SAP Security Configuration and Deployment: The IT Administrator's Guide to Best Practices*. Burlington, MA: Syngress Publishing.

ITGI (2006). *IT Objectives for Sarbanes-Oxley*. Rolling Meadows IL: IT Governance Institute.

KPMG (2004). Fraud Survey 2004 Retrieved 16/04/2007, from http://www.kpmg.com

KPMG (2007). Profile of a Fraudster Survey 2007 Retrieved 18/03/2009, from http://www.kpmg.com

KPMG (2008). Fraud Survey 2008 Retrieved 03/11/2009, from http://www.kpmg.com

KPMG (2009). Fraud Survey 2009 Retrieved 18/01/2010, from http://www.kpmg.com

Kuhn Jr, J. R., & Sutton, S. G. (2010). Continuous Auditing in ERP System Environments: The Current State and Future Directions. *Journal of Information Systems, 24*(1), 91-112.

Kuhn, J. R., & Sutton, S. G. (2006). Learning from WorldCom: Implications for Fraud Detection through Continuous Assurance. *Journal of Emerging Technologies in Accounting, 3*(1), 61-80.

Lager, M., & Tsai, J. (2008). SAP Retains Market-Share Lead in CRM. *Customer Relationship Management*(October 2008), 17-18.

Lanza, R. B. (2003). *Proactively Detecting Occupational Fraud Using Computer Audit Reports.* Florida: The IIA Research Foundation.

Li, N., Tripunitara, M. V., & Bizri, Z. (2007). On mutually exclusive roles and separation-of-duty. *ACM Trans. Inf. Syst. Secur., 10*(2), 5.

Little, A., & Best, P. J. (2003). A framework for separation of duties in an SAP R/3 environment *Managerial Auditing Journal 18*(5), 419-430.

Narayan, V. (2008). *Financial Accounting (FI). SAP FI/CO Questions and Answers.* Sudbury: Infinity Science Press.

NIST (2005). An Introduction to Computer Security: The NIST Handbook, Special Publication 800-12 (Vol. 800-12, Available from http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf

O'Gara, J. D. (2004). *Corporate Fraud Case Studies in Detection and Prevention*: Wiley.

Padhi, S., N (2010). *SAP ERP Financials and FICO Handbook*. Sudbury: Jones and Bartlett.

Potla, L. (2003). Detecting Accounts Payable Abuse Through Continuous Auditing. *ITAudit, 6*(3). Retrieved from http://www.theiia.org/itaudit/index.cfm?fuseaction=forum&fid=5458

PwC (2009). The Global Economic Crime Survey. Economic crime in a downturn. November 2009 Retrieved 08/02/2010, from http://www.pwc.com/gx/en/economic-crime-survey

Rezaee, Z., Sharbatoghlie, A., Elam, R., & McMickle, P. L. (2002). Continuous Auditing: Building Automated Auditing Capability. *Auditing, 21*(1), 147.

SAP-AG (2009). SAP Library. Retrieved 23 March 2010, from SAP AG: http://help.sap.com/erp2005_ehp_04/helpdata/EN/e1/8e51341a06084de10000009b38f83b/frameset.htm

SAP (2010). SAP Named Worldwide Market Share Leader in Business Intelligence, Analytics and Performance Management Software by Top Industry Analyst Firm Retrieved 27/10/2010, 2010, from http://www.sap.com/australia/search/index.epx?q1=fraud+detection&num=10

Singleton, T., Singleton, A., Bologna, J., & Lindquist, R. (2008). *Fraud Auditing and Forensic Accounting*: John Wiley & Sons.

Srinidhi, B. (1994). The Influence of Segregation of Duties on Internal Control Judgments. *Journal of Accounting, Auditing & Finance, 9*(3), 423-444.

Standards Australia (2008). Australian Standard AS 8001-2003 - Fraud and Corruption Control Retrieved 15/01/2010, from http://www.saiglobal.com/shop/Script/search.asp

Tackett, J. A. (2007). Digital analysis: A better way to detect fraud. *Journal of Corporate Accounting & Finance (Wiley), 18*(4), 27-36.

Tatum, M. (2010, 08 September 2010). What is an Audit Trail Retrieved 11/112010, 2010, from http://www.wisegeek.com/what-is-an-audit-trail.htm

Vasarhelyi, M. A., Alles, M. G., Kogan, A., & O'Leary, D. (2004). Principles of Analytic Monitoring for Continuous Assurance. *Journal of Emerging Technologies in Accounting, 1*, 1-21.

Vasarhelyi, M. A., & Halper, F. B. (1991). The Continuous Audit of Online Systems. *Auditing, 10*(1), 110-125.

Weber, R., A (1999). *Information Systems Control and Audit.* Upper Saddle River, NJ: Prentice Hall.

Wells, J. T. (2002). Let Them Know Someone's Watching. *Journal of Accountancy, 193*(5), 106-110.

Wells, J. T. (2002a). Billing schemes, part 1: Shell companies that don't deliver. *Journal of Accountancy, 194*(1), 76-79.

Wells, J. T. (2003). Protect small business. *Journal of Accountancy, 195*(3), 26-32.

Wells, J. T. (2008). *Principles of Fraud Examination* (2nd Ed. ed.): John Wiley & Sons

ooOoo