Enterprise Resource Planning (ERP) is an enterprise-wide information system designed to coordinate all the resources, information, and activities needed to complete business processes such as order fulfillment or billing. Many firms rely on ERP systems to implement business processes and integrate financial data across their value chains. This reliance increases the importance of ERP system security in protection of a firm's information assets. In recent years, the audit of ERP security has gained importance and begun receiving an increasing percentage of firms' audit budgets. However, the audit of ERP security remains a complex, lengthy and costly task due to a confluence of factors.

ERP systems are inherently complex systems spanning many functional areas and processes along a firm's value chain. They are designed to provide flexible solutions to business problems. The sheer number of possibilities available for configuring an ERP system implies many potential security configurations. However, ERP systems pay little attention to potential conflicts and problems in those security configurations. Deployment and implementation of ERP systems also pay little attention to security implications, as the main purpose is to solve business problems within time and budget. In post implementation stages, auditors have access to rudimentary ERP tools and capabilities for auditing security configurations. There are also shortages of staff members trained in the ERP security.

Unfortunately, the increased enthusiasm on this subject has been met with complex and costly challenges. Many companies and audit firms are not yet prepared to tackle the need for a rigorous ERP security audit. Major challenges in auditing ERP Security are given as follows:

- **Complexity of ERP systems:** Complexity of ERP systems leads to security vulnerabilities. ERP systems must be able to process a wide array of business transactions and implement a complex security mechanism that provides granular-level access to users. For example, in SAP R/3, hundreds of authorization objects are used to allow access to various actions in the system. A small or medium-sized organization may have 100 transactions that are commonly used, and each transaction typically requires at least two authorization objects. If the company has 200 end users who fill a total of 20 different roles and responsibilities, there are approximately 800,000 (100*2*20*200) ways to configure security in the ERP-and this scenario excludes other complexity factors, such as multiple transactions sharing the same authorization objects, an authorization object having up to 10 fields that can be assigned to various values, and the possibility of using position-based security. The point of this illustration is that the inherent complexity of an ERP system increases the complexity of security configurations and leads to potential security vulnerabilities. Flaws, errors and Segregation-Of-Duty (SOD) conflicts become more likely. Consider a scenario in which a security administrator has to grant read-only access to transaction X, which requires him/her to assign 10 authorization objects to the role. At a later point in time, management decides to grant write access to transaction Y, which implies assigning five more authorization objects. One of the objects is common to both transactions and determines the write capability. Although these two changes are seemingly independent, due to the shared authorization object granting write privileges, the unintended consequence is a potential SOD conflict. An ERP system does not automatically check for these kinds of security vulnerabilities. Unless the security administrator is well trained and employs rigorous positive and negative testing, he/she is likely to miss the unintended consequence of allowing write access to both transactions X and Y. As the number of potential configurations and authorization objects increases, it becomes increasingly difficult and costly to analyze the security implications of ERP configurations, such as the unintentional creation of SOD conflicts.

- **Lack of ERP Tools:** ERP tools for security audit are inadequate. Most of the security tools available in ERP packages are not designed to facilitate efficient and effective audit of ERP security. The main emphasis of ERP tools is on security configuration and maintenance. Recently, there has been an increase in the number of third-party product offerings assisting with ERP security and SOD reviews. However, many users complain that those tools often generate false positives and create more work for auditors.
- **Customization of ERP Systems:** The customization of ERP systems to firms inhibits the development of standardized security solutions. Every ERP implementation contains some level of customization specific to the firm undertaking the implementation. However, customization makes it difficult to develop a standard approach or methodology for conducting ERP security audits.
- **Manpower:** There is a shortage of manpower trained in ERP security**.** Most ERP training programs are designed for implementation efforts. They offer very little on ERP security and audit. Thus, there is a shortage of auditors who are trained in ERP security.
- **Inadequate attention towards security:** Implementers pay inadequate attention to ERP security during deployment. Many companies do not pay adequate attention to security implications of ERP configurations

during the deployment and implementation of ERP systems. Implementation teams are usually tasked with finishing the implementation projects on time and within budget. They do not pay adequate attention to security implications since it increases implementation time and budget. Due to limited emphasis on security implications, ERP security becomes too lax, making post implementation problem identification and remediation very costly.

- **Conventional Approach:** Most ERP security audits today are performed using a manual approach. There is little automation beyond the use of native tools that come standard with ERP packages. Unfortunately, the bottleneck of the manual approach is the limitation of the native security reporting tools found in most ERP products. These native tools are not designed to facilitate a large-scale audit effort, but rather to help security administrators perform occasional validation of the accuracy of security configuration. They allow reporting on only a single transaction per query, which may be adequate for a security administrator who works full time and handles each transaction request individually; however, it is not as practical for an IT auditor who is expected to perform the audit in a limited period of time and must test a large number of transactions. Although some IT auditors are able to utilize technology to perform this process more efficiently than others, as long as the process is based on the same philosophy of manual extraction followed by analysis, it continues to be an incredibly tedious and time-consuming task. The manual method is also prone to human errors.

Conclusion:

In today's business life, ERP is recognized as an effective tool which supports most of the business systems that maintain the data needed for a variety of business functions such as Manufacturing, Supply Chain Management, Financials, Projects, Human Resources and Customer Relationship Management in a single database. On the other hand, auditing of ERP security is also a demanding area which requires proper attention. Though many steps have already been taken by various researchers world wide, but for smooth and efficient functioning of business tasks in a better manner, there is still a need of many more initiatives to be taken in this direction.

Sources:

[1]. Roberta S. Russell, A Framework for Analyzing ERP Security Threats, www.cimap.vt.edu/CIIA/Papers/Session1-4-Russell.pdf

[2]. www.net-security.org/article.php?id=691

[3]. www.csoonline.com/article/216940/The_ERP_Security_Challenge