One element of IT audit is to audit the IT function. While probably more common in external audit, it certainly could be a part of internal audit, especially in a risk assessment activity or in designing an IT function. While there are many important aspects of the IT function that need to be addressed in an audit or risk assessment, one is undoubtedly proper segregation of duties (SoD), especially as it relates to risk. Similar to traditional SoD in accounting functions, SoD in IT plays a major role in reducing certain risk, and does so in a similar fashion as well. This article addresses some of the key roles and functions that need to be segregated.

**IT Duties vs. User Departments**

The most basic segregation is a general one: segregation of the duties of the IT function from user departments. Generally speaking, that means the user department does not perform its own IT duties. While a department will sometimes provide its own IT support (e.g., help desk), it should not do its own security, programming and other critical IT duties. To mix critical IT duties with user departments is to increase risk associated with errors, fraud and sabotage. User departments should be expected to provide input into systems and application development (i.e., information requirements) and provide a quality assurance function during the testing phase. In fact, a common principle of application development (AppDev) is to ask the users of the new application to test it before it goes into operation and actually sign a user acceptance agreement to indicate it is performing according to the information requirements. However, the majority of the IT function should be segregated from user departments.

**Database Administrator vs. Rest of IT Function**

The database administrator (DBA) is a critical position that requires a high level of SoD. The DBA knows everything, or almost everything, about the data, database structure and database management system. Thus, this superuser has what security experts refer to as “keys to the kingdom”—the inherent ability to access anything, change anything and delete anything in the relevant database. This situation leads to an extremely high level of assessed risk in the IT function.

Because of the level of risk, the principle is to segregate DBAs from everything except what they must have to perform their duties (e.g., designing databases, managing the database as a technology, monitoring database usage and performance). The IT auditor should be able to review an organization chart and see this SoD depicted; that is, the DBA would be in a symbol that looks like an island—no other function reporting to the DBA and no responsibilities or interaction with programming, security or computer operations (see figure 1).

A similar situation exists for system administrators and operating system administrators.

**APPDEV vs. DBA and IT Operations**

The development and maintenance of applications should be segregated from the operations of those applications and systems and the DBA. That is, those responsible for duties such as data entry, support, managing the IT infrastructure and other computer operations should be segregated from those developing, writing and maintaining the programs. The same is true for the DBA.

It is also true that the person who puts an application into operation should be different from the programmers in IT who are responsible for the coding and testing.

This SoD should be reflected in a thorough organization chart (see figure 1).
NEW APPDEV VS. APP MAINTENANCE

For organizations that write code or customize applications, there is risk associated with the programming and it needs to be mitigated. One way to mitigate the composite risk of programming is to segregate the initial AppDev from the maintenance of that application.

In a large programming shop, it is not unusual for the IT director to put a team together to develop and maintain a segment of the population of applications. For instance, one team might be charged with complete responsibility for financial applications. This situation should be efficient, but represents risk associated with proper documentation, errors, fraud and sabotage.

This scenario also generally segregates the system analyst from the programmers as a mitigating control. However, this control is weaker than segregating initial AppDev from maintenance.

The above scenario presents some risk that the applications will not be properly documented since the group is doing everything for all of the applications in that segment. This is especially true if a single person is responsible for a particular application. Improper documentation can lead to serious risk. For example, if key employees leave, the IT function may struggle and waste unnecessary time figuring out the code, the flow of the code and how to make a needed change. Documentation would make replacement of a programmer process more efficient.

The lack of proper SoD provides more opportunity for someone to inject malicious code without being detected—because the person writing the initial code and inserting malicious code is also the person reviewing and updating that code. Therefore, a lack of SoD increases the risk of fraud. If the departmentalization of programmers allows for a group of programmers, and some shifting of responsibilities, reviews and coding is maintained, this risk can be mitigated somewhat.

A similar situation exists regarding the risk of coding errors. If the person who wrote the code is also the person who maintains the code, there is some probability that an error will occur and not be caught by the programming function. This risk can be somewhat mitigated with rigorous testing and quality control over those programs.

A proper organization chart should demonstrate the entity’s policy regarding the initial development and maintenance of applications, and whether systems analysts are segregated from programmers (see figure 1).

INFORMATION SECURITY VS. REST OF IT FUNCTION

Much like the DBA, the person(s) responsible for information security is in a critical position and has “keys to the kingdom” and, thus, should be segregated from the rest of the IT function. This person handles most of the settings,
configuration, management and monitoring (i.e., compliance with security policies and procedures) for security. Login credentials may also be assigned by this person, or they may be handled by human resources or an automated system. Therefore, this person has sufficient knowledge to do significant harm should he/she become so inclined. This risk is especially high for sabotage efforts.

**AUDITING THE IT FUNCTION AND SOD**

The audit program should include:

- A review of the information security policy and procedure
- A review of the IT policies and procedures document
- A review of the IT function organization chart (and possibly job descriptions)
- An inquiry (or interview) of key IT personnel about duties (CIO is a must)
- A review of a sample of application development documentation and maintenance records to identify SoD (if in scope)
- Observation of personnel for SoD
- Verification of whether maintenance programmers are also original design application programmers
- A review of security access to ensure that original application design programmers do not have access to code for maintenance

**CONCLUSION**

Figure 1 summarizes some of the basic segregations that should be addressed in an audit, setup or risk assessment of the IT function. The sample organization chart illustrates, for example, the DBA as an island, showing proper segregation from all the other IT duties. The same is true for the information security duty. The AppDev activity is segregated into new apps and maintaining apps. IT auditors need to assess the implementation of effective SoD when applicable to audits, risk assessments and other functions the IT auditor may perform. The reason for SoD is to reduce the risk of fraud, ( undiscovered) errors, sabotage, programming inefficiencies and other similar IT risk.