Metrics for Due Diligence

"Due diligence" is a common term in financial business transactions. When applied to computer security, it refers to the protection of assets stored in a computer. The practice of due diligence requires an Information Technology (IT) organization to maintain an "industry standard" level of computer security. Where an assessor is chartered to ascertain security at a given entity, due diligence requires an attempt in good faith to *assess* the security at the entity's site. Such *assessment* follows *standards* that the measuring organization thinks appropriate to compare to an IT infrastructure. To measure security, it must first be possible to define those standards, that is, to define *standards of due care* with respect to computer security. Once those standards are established, it will be possible to establish *verification mechanisms* that measure compliance with the standards. *Verification mechanisms* provide *security metrics*.

Historical Approaches

All models for measuring security implicitly adopt standards of due care. The security measurement is typically performed by an "evaluator," or a person who compares the actual state of an IT infrastructure to the standard. Currently used computer security assessment models fall into one of five categories, each with its own set of assumptions and metrics (see Table 1).[1]

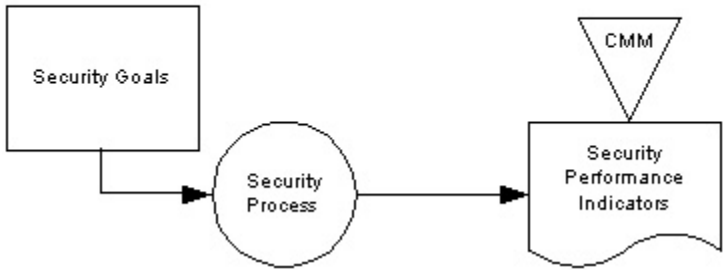
Table 1: Security Assessment Model Comparison

Assessment Model Type:	Utilizes as evaluators:	Evaluators' standards consist of…	Measurement activity consists of…	There is a tacit assumption that…	Metrics produced are…
Capability Maturity	consultants and quality analysts	…process requirements	…identifying the process by which security is achieved, then determining the extent to which the process is mature (for example, is it quantifiable and/or does it continuously improve).	… organizations committed to securing their infrastructure will formally adopt a process for so doing.	…organizational ratings.
External Audit	Independent Third Parties	…best practices	…comparing the level of management's control over the current systems infrastructure to that which would result if best practices were followed.[2]	… there are "best practices" available on how to secure a given infrastructure.	…vulnerability listings.
Internal Audit	Internal reports to the Board of Directors	… control objectives set by management	…determining the extent to which management-defined controls are appropriate and if in fact they are followed.	… management has adopted a set of control objectives designed to secure information systems assets.	…vulnerability listings.
Risk Analysis	accountants	…dollars spent in like organizations	…comparing the dollar value at risk from potential harm to a system to the cost of implementing security.	… there is a known dollar figure that represents how much it would cost to "completely secure" the IT infrastructure.	… spending recommendations.
Automated Verification	technologists	…maximally restrictive configuration parameters	…quantifying measurable parameters inherent in an IT infrastructure that reflect its "security profile."	… there are specific measurable parameters inherent in an IT infrastructure that reflect its "security profile."	…summary of measured variables.

[1] Table is a summary of: Bayuk, Jennifer, *Security Metrics*, The Computer Security Journal, Vol XVII, No 1, 2001.

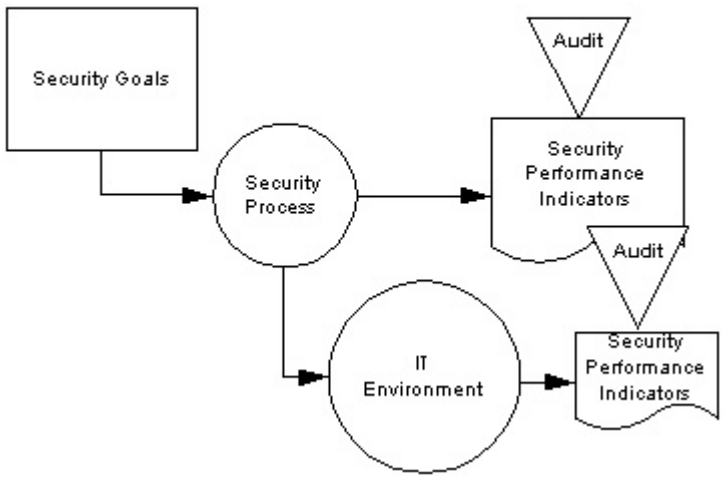
Historically, a standard of due care with respect to computer security has been interpreted as management control over information system processing. Management control means that "what management desires and expects to be the case" is in fact "the case." Management is assumed to desire and expect protection of information systems assets. If management does not have direct control over how information is processed, it is assumed that there is a high risk that information systems assets will be compromised. This logical approach leads to the assumption that if management exercises due care, there will be processes in place that ensure security is performed adequately. This interpretation is directly reflected in the practice of "Capability Maturity" assessments. It is also directly reflected in legislation such as Sarbanes-Oxley. The following figure demonstrates this approach:

Figure 1



More recently, it has become common for management to outsource data handling. Yet, to demonstrate due diligence, it still must ensure that external entities properly safeguard sensitive data. This has led to the proliferation of "third-party" audits, wherein management hires an audit firm to determine whether another company has processes in place designed to effect information security.[3] These "independent: measurement strategies rely on standards like "best practices" and "management controls" as the basis for independent examination. They rely on human auditors to provide assessments based on their experience with such processes in other organizations. They evaluate the entity's "process" for delivering security, then add a few "spot checks" to see if a system configuration dictated by the process is in fact in place. Both Internal and External Auditors follow the approach, which is demonstrated in Figure 2.

Figure 2

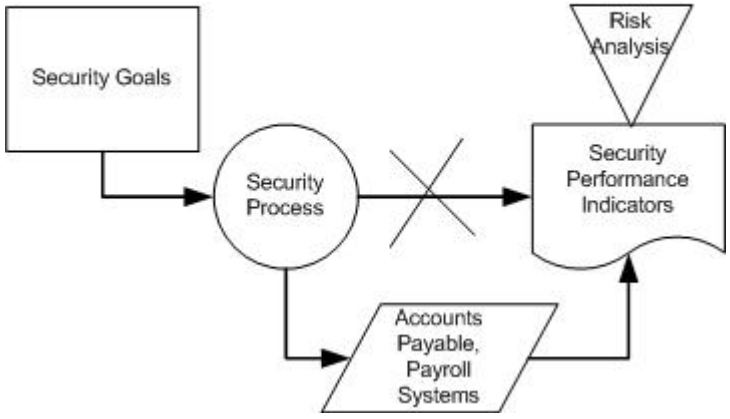


[2] The audit firm is thus the "third-party." Because one obvious best practice is managing security to keep out unwanted intruders, penetration studies fall under this model.
[3] For example, SAS70 and SysTrust, as well as external penetration reviews.

As smaller and smaller companies vie for larger and larger chunks of outsourcing business, the dependence on these "best practice" standards becomes harder and harder to justify. Best practices typically involve lots of documentation, as well as checks and balances. A ten-person IT department is unlikely to have best practices or even detailed management control objectives with respect to IT. A "reasonableness" standard indicates that adoption of "best practice" management processes should not be necessary to achieve security in a small shop. How then can a very large multinational bank gain assurance that security is in fact in place at their ten-person application development shop? At the current time, this is a rhetorical question.

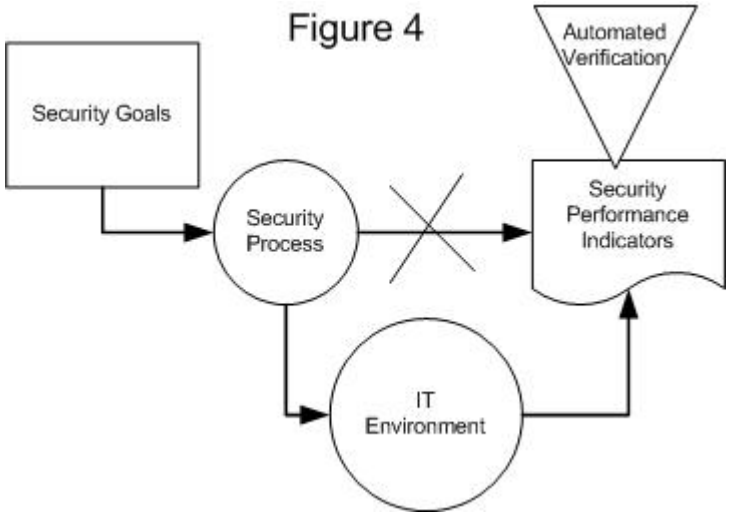
One approach that has been suggested, though usually as a straw man,[4] is a financial risk analysis. The approach assumes that the extent to which security is effective can be measured in terms of the dollar spend on security. This is usually calculated as a percentage of information technology total dollar spend. It is common for IT Security surveys to publish "security spend" data with respect to an industry or with respect to the size of an IT organization. However, this standard has no foundation in the extent to which security mechanisms are actually implemented, and so must be discounted. It is mentioned only because it has a surface "common sense" appeal, and is reflected in the practice of security "Risk Analysis" security assessment approaches. It is also reflected heavily in security vendor marketing material. The following figure demonstrates the approach:

Figure 3



Of the five security assessment methods described in Table 1, the only one that does not rely totally on subjective process definition and subsequent subjective evaluation is the Automated Verification approach. This security assessment technique assumes that the standard of due care in security is a technical lockdown of system parameters. The result of the lockdown is that all access to data or programs corresponds to legitimate use cases. Weak security is evident if that there is some system parameter that has not been purposely configured with security in mind, and thus allows more systems access than is required for the system to fulfill its function. This definition is directly reflected in legislation such as Grahm-Leach-Bliley and California Civil Code 1798 (a National Version pending as Senate Bill 1750). Security process is required to secure the IT environment, but the only indicator that security processes are working is that the environment is in fact secure. Where defects are found, processes should be improved, and the environment measured again. Figure 4 demonstrates the approach.

[4] A "strawman" approach is one that is discussed only in hope that the discussion will yield a better approach.



The Twenty Question Method

Recently, the quest for due diligence in verifying third party application security has resulted in a plethora of questionnaires wherein one company asks written questions of another in an attempt to assess their information security environment.[5] This "Twenty Question" method of security verification is based on universal agreement that it is a good idea to have a very simple rating systems for security wherein service providers can be given ratings based on the security processes they have in place. However, without any verification that the answers to the questions are correct, this "diligence" amounts to no more than a gentlemen's agreement. In the absence of verification mechanisms, there is only the threat of legal ramifications if answers are found misleading. Were the control a login screen rather than a list of questions, security professionals would refer to this type of enforcement mechanism as, "keeping your friends out."

To truly impose security requirements on a third party, one must provide a bridge from the fairly broad, generally accepted criteria of what counts as good security to the verification of the existence of the criteria with a set of automated tests. Only then will there be a foundation with which to move the "Twenty Question" method to the required rating system. Let us for the purposes of exploring the question assume that there is a list of objective criteria, that, if met, would guarantee that a given systems environment is secure. Those criteria could be tested in many different ways. One way to test would be to ask the corresponding systems' management if the criteria are met, albeit a very weak test. A progressively more stringent test would be to review the process documentation that management uses to run the systems environment. Like a CMM Auditor, an assessor could verify that if the process was running correctly, the criteria would be met. A third method would be to spot check the process like an Internal Auditor would. A fourth method might be to try to break a control that is claimed to in place like an External Auditor might (i.e. penetration test). However, the only real way to know that the criteria is met is to have an automated test that verifies it.

Table 2 outlines a security criterion and corresponding verification methods. Note that listing the automated verification method makes it easier to come up with objectives for documentation review and manual testing. Also note that penetration testing is recognizably the most difficult of the available options. Where security verification is performed with an automated approach as the end goal, control objectives are clear and it is possible to assess whether processes are followed without even reviewing them. Moreover, if processes were not followed, but management somehow achieved the goal of restricting the LAN to employees anyway, the only judgment that can be made is that management may not have chosen the correct processes. There is no penalty if they are nevertheless secure.

[5] As a Chief Information Security Officer in a Wall Street Clearing Firm, I have been subjected to five of these questionnaires in the past two months, all from Fortune 500 financial institutions.

Table 2: Security Assessment Progressive Approach					
Criterion	Ask Manage-ment	Review Documenta-tion	Spot checking	Penetration Test	Automated Verification
All users of the LAN are employees.	question -naire	Review procedures by which LAN IDs are added or removed.	Take a statistical sampling of LAN IDs. Look them up in the payroll system.	Try to guess LAN user names and passwords using user names that do not correspond to employees.	As part of the process to add a user, store the unique index from the payroll system in LAN user record, run a program to verify that these index match active payroll records.

One criticism of this approach is that an assessor may find all records in place the day of the verification but no be assured that it will not be in place the next day. The obvious rebuttal to that criticism is that the same can be said with respect to a process audit or any other verification of any management quality initiatives.

Clearly this approach combines the best features of all the security assessment models. Labeling the criteria as "verification levels." Following the columns of Table 2, the levels are described as:

1. Management Asserts that Criteria is Met
2. A Process is Documented that, if followed, would meet Criteria
3. Statistical Sampling of Control Points in Documented Process Indicate that Criteria is Met
4. Attempts to establish that the criteria is not met fail
5. An agreed upon automated process verifies that Criteria is Met

Clearly, an organization that meets security criteria at level 5 is more secure than one that simply answers questionnaires and does not submit to automated tests. Just as clearly, the whole concept of the levels relies on universal acceptance of the original criteria as well as the fact that automated verification methods exist that test for the criteria. Many types of security criteria exist, and the foundation for this approach must be based on a generally accepted criteria with generally accepted verification approaches.

The only globally accepted organization that accredits information systems auditors is the Information Systems Audit and Control Association (ISACA). It operates in over a hundred countries on five continents and has roughly 30,000 members, most of them Certified Information Systems Auditors, and it has recently introduced a certification in Security Management (CISM). ISACA is affiliated with a nonprofit research foundation whose objective it to research, develop, publicize and promote an authoritative, up-to-date, international set of generally accepted information technology control objectives for day-to-day use by business managers and auditors. Its publications cover the control environment from the multiple perspectives, from line management to the Board of Directors. At the same time, they are consistent with each other. It covers as many process measures and security levels as any security CMM. Its recommendations for implementing IT Security cover and extend BS17799. Its Management Process Goal and Performance Indicators cover ITIL. Moreover, it is the only published standard that provides directions for measuring the security of the systems environment in a standard way. It thus provides a verification standard upon which any standard of security process improvement may be mapped.

ISACA's Control Objective "Ensure Systems Security" includes 20 control practices:[6]

1. Identification, Authentication, and Access
2. Security of Online Access to Data
3. User Account Management
4. Management Review of User Accounts

[6] It also includes one called "Manage Security Measures." It was omitted from the verification lists because if the other twenty control practices are in place, it may be assumed that someone is managing the process.

5. User Control of User Accounts
6. Security Surveillance
7. Data Classification
8. Central Identification and Access Rights Management
9. Violation and Security Activcity Reports
10. Incident Handling
11. Reaccreditation
12. Counter-Party Trust
13. Transaction Authorization
14. Nonrepudiation
15. Trusted Path
16. Prtection of Security Functions
17. Cryptographic Key Management
18. Malicious Software Prevention, Detection and Correlation
19. Firewall Architectures and Connections with Public Networks
20. Protection of Electronic Value

An audit of the high level control objective, ensure systems security, must include verification of these twenty control practices. These practices can be verified by asking questions, by reviewing documentation, by spot checking, penetration testing, or automated testing. The challenge is to come up with automated tested that provide closure for the basic security requirements met by each control practice. Once the basic requirements and automated tests are laid out, auditors whose verification mechanisms which cannot be automated will at least have clear and consistent objectives as they ask questions, review documentation, and spot check. The clear and consistent objectives are of course prerequisites for a rating system.

An Automated Verification Standard

Of course, standards of due care defined by the automated verification approach currently change with the security technology being measured. This is because what counts as a securely configured system parameter must be defined in advance of measurement. To define in a technology-independent way what counts as a secure configuration requires generalization about what mechanisms must be in place to provide adequate security. Such generalizations exist, but not in the form of objectively defined standards that have independent automated verification mechanisms.

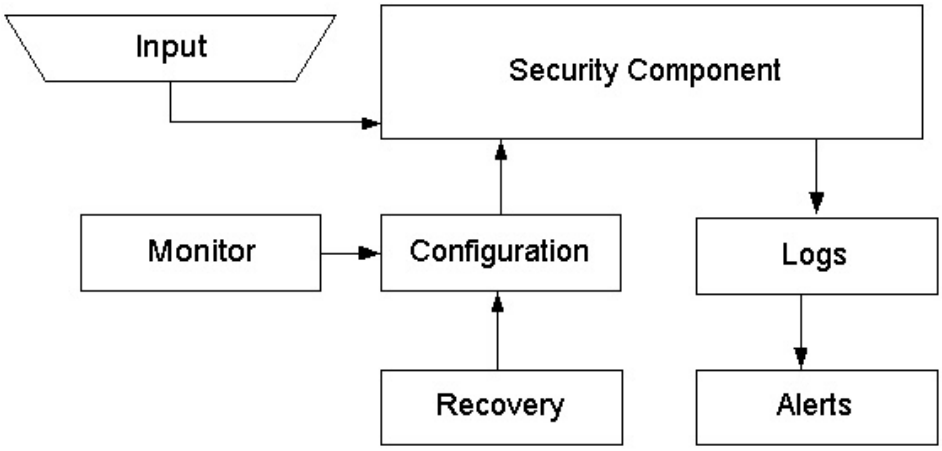
Nevertheless, the very dissimilarity among vendor and application security mechanisms is the strongest case for adopting a standard of due care that can be measured using an automated approach. It is common knowledge among technologists that the IT audit assessments are easy to subvert. Auditors will test what they are told is germane to the security of the application or site they are auditing. However, they believe what they are told about the infrastructure. Note that audit requirements are in business terms. Auditors come in looking for the “Financial Processing” system, not the “shared application services server.” If an IT Manager can persuade the auditor that the “Financial Processing” system does not rely on the “shared application services server” for its security, that system may be left entirely out of the audit, though it may present significant vulnerabilities to the infrastructure. A standard, measurable method of configuring and measuring security would presumably uncover such dependencies and help define scope.

Moreover, dissimilarity among vendor and application security mechanisms should not immediately discount a standard approach to automated computer security assessment. Each system security mechanism is in turn a system that is supported. Figure 5 demonstrates that, at a high enough level, security mechanisms are similar. All have some type of user input or administrative processes. All have a configuration that is either read upon startup or can be changed on the fly, and that configuration is stored. A security mechanism’s configuration should be monitored for integrity and it should be automatically recoverable in the event of a disaster. Security mechanisms are also similar in that they log events that are thought by their designers to have significance to the security of the environment, and that log may be used to produce security alerts.

These standard security components reflect standard requirements. All security standards have roots in the nearly half-century-old mantra, “prevention, detection, recovery.” That is, security first and foremost is the process of preventing harm from happening to systems. However, because systems must serve some useful

purpose, they must allow some external input. Because it is an incomputable question whether access control, or *prevention* systems will always be effective, we must also have mechanisms to *detect* when they fail. Where detection methods also fail, we face system corruption. If both our prevention and detection mechanisms fail, we had better be able to *recover*. Though it will sometimes be articulated as P*revention, Detection, Correction*, or some other such variation, *Prevention, Detection, Recovery* is the basis for all security training and certification programs, and a mantra to which all security professionals subscribe. Thus, all security mechanisms have a configuration that aims to prevent harm. These are monitorable and recoverable. They have user and/or administrator input. They have logs that may produce alerts.

Figure 5: Security Mechanism Architecture



Moreover, there already exist standard ways to test these standard mechanisms. Although the vast majority of time in an audit is spent in interviewing people, reading procedures, or reviewing process, auditors know that only the actual system configuration counts as evidence that that prevention, detection, and recovery mechanisms are in place. Hence, almost every audit plan has at least one step devoted to "on-line testing." Usually, a lead auditor will call in a "technical expert" to perform this step in the audit. The expert logs into the system in question and looks at system parameters whose values indicate that standard security mechanisms are in place. After a few days spent in the bowels of the computer, the expert publishes a bullet list of security parameters that do not meet "best practices."

Auditors routinely start any audit of any operating system by running configuration collection utilities that gather evidence of "preventive" controls on the infrastructure. Most large audit firms have automated tools that further standardize the approach within their firm. These files provide evidence that the standard security components are configured correctly. Table 3 shows what system parameters are examined in different products in order to provide evidence that input mechanisms are configured properly. It is true that a standard approach an auditor uses to examine a Microsoft NT Server will not work for an IBM mainframe. Different vendors implement even extremely basic security, such as computer login, differently. So security verification processes have to be different by platform. However, if we expect to independently and objectively measure due care with respect to a automated verification standard, we must devise ways to automatically test similar security mechanisms is a standard way. The problem is solvable to the extent that security mechanisms are similar and similarly verifiable.

Table 3: Security Input Component Measurement Example for Various Products

Product	Security Configuration Parameter	Standard	Measurement
E-Trust for Solaris	Seosdb list of users	employee database using employee number as	percentage users mapped
IBM MVS	Top Secret user listing		

Microsoft AD	SAM	unique identifier and department number as evidence of group authorization	
Remote Access equipment configured for use with SecurID	Listing of valid users from the ACE Server		
Tripwire	Those operating system users that can update the config files		

There is no technical reason why external auditors cannot share just one standard way of automating the evaluation of this security component configuration. A great deal of time is spent in every audit going over lists of users and comparing that list with people in a given department or those who have authorization forms on file. Suppose that there was instead a best practice of keeping unique identifiers for all users in a given company? This would allow operating systems and applications to store the unique identity of the user and that string or number could be verified against a central store. The standard would be a predefined representation of user identity in the form of a system parameters. Suppose major operating system vendors could provide user and group files in a standard way, using an index that corresponded to a platform-independent digital identifier? Then auditors could take standard files from operating systems and HR systems and run standard software that would produce a metric on the number of users that fell outside authorized system access groups. An automated verification security assessment approach will thus drive best practices in the direction of being independently verifiable.

Implementation Ideas

With the assumption that we can automate the assessment of computer security comes an idea for implementation. If there existed a program that gathered configuration files from different operating systems and displayed them in the same format, this would provide a standard for assessing security of different platforms in the same automated way. This approach is common when it comes to interpreting logs from different security products. There are several vendors that currently offer "log parsing" products that "normalize" log data across different security vendor platforms.

That is to say, in the current Information Security product development environment, it actually seems feasible to pay one security vendor to parse log data produced by other security vendors. Applying the same approach to measuring preventive controls is also feasible. We could pay vendors to parse other vendor's security configuration files as well. But consider the number of platforms that a given configuration format script would have to support in order to be useful in all infrastructures, and the amount of translation between what constitutes network access control measurements on an NT Server versus an IBM Mainframe. Imagine the endless software updates and version compatibility issues with the existing security log parsing systems, and extend that to possible security configuration verification systems. Consider the time and expense not only on the part of the vendor, but also on the part of the client performing the upgrades. Moreover, consider that a bug in a security vendor information consolidation product could have adverse security implications for all clients.

Suppose instead that all makers of computer security products incorporate features that allow the security configuration of their products to be displayed in a way that is operating system independent. Then the security of any computer could be verified by an automatically measurable due care standard. External auditors have long known the possible number of ways that certain operating systems can be configured to ensure the highest possible level of access control. Displaying this information in a standard format that allows for independent measurement could reduce security metrics to grunt work and allow a realistic application of the automated verification model of security assessment.

For a simple proof of concept, assume that a given company utilizing the progressive approach in Table 2 has a pure Microsoft Active Directory environment and the control objective to be tested is ISACA's Identification, Authentication, and Access. Figure 6 outlines the automated verification process as it would work in that environment.

For another example, note that many security processes rely on the fact that object code delivered to a production computer is the same as that in the test computer. If there was a standard way to compare binary files across platforms, then auditors could, with a simple network interface, verify that code in test

matched code in production to the percentage of matching binaries or even matching bits. This approach has already been validated and published by the File Signature Database Coalition (FSBD).[7]

Conclusion

Current security assessment practices of evaluating security processes are not adequate to the task of verifying systems security. There is instead an urgent need for an objective measurable due care standard that does not rely on organizational process or third party evaluation. Computer security measurement should be able to verify not just that management is in control, but that data is "safe."

The practice of due diligence requires an Information Technology organization to maintain an "industry standard" level of computer security. This paper describes a methodology that would allow a predefined industry standard to be *automatically measured*. Note that anything *assessors* would be able to do with these automatically-measured *due care standards* could also be used by *security personnel* within companies. Security departments would be able to use these *security metrics* and thus measure the quality of their own security. The techniques could directly support internal security quality initiatives (e.g. Six Sigma). The metrics provide objective verification that due diligence has been followed.

Figure 6: Automated Verification via Microsoft

To Verify: All active accounts in an AD Domain correspond to authorized user records.

Setup required:

- Digital Identity Source (*DIS*) A database of record on authenticated users, e.g. Payroll, Account Master, Customer Relationship Management System. The only requirement for this database is that it have a *unique index* and a *status flag*.
- Digital Identity Index (*DII*) A Customized field in Active Directory that only allows input in the same format as the *unique index* in the DIS.
- Identity Type List (*ITL*) A list of *status flags* in the DIS that corresponds to authorized users. For a payroll system, this would be the same as those to whom checks were written. For a Customer Relationship Management System, this may be an "active" flag, indicating that the customer is current.

Algorithm for Processing:

- Set two numeric variables to zero: *authorized* and *unauthorized*.
- For *each record* in the AD, check the *DII* against the *unique index* of the *DIS*. If the record exists in the *DIS*, check its *status flag* against the *ITL*. If the *status flag* is included in the *ITL*, then increment *authorized* by one. . If the *status flag* is not included in the *ITL*, then increment *unauthorized* by one.
- Divide *authorized* by the sum of *authorized* and *unauthorized*.
- Convert result to a percentage

[7] http://www.tripwire.com/fsdb/index.cfm