Global Information Assurance Certification Paper

Garren Shannon
GSEC Practical Assignment V 1.4b
Security is Free: Implementing a Continuous Security Improvement (CSI) program.
Jan 14, 2003

Abstract:

One of the most challenging aspects of implementing a security program is where to start. Security is a much larger project than simply a firewall project or scanning the systems for vulnerabilities. The question I am trying to answer is "How do I get the entire company moving towards secure systems and practices all at the same time?" I need to look at the network infrastructure as well as servers, the people who manage the servers, the end users, policies… etc. I need to involve everyone in this program… but how? Once I get them into the game, how do I engage them in continuously improving the program? One method, which to my knowledge has not yet been applied to this type of program, is to modify a Total Quality Management (TQM) program to fit a security program.

TQM builds end user feedback into the program. It is, by design, an enabler for allowing information to flow up from end users and down from administrators. It also provides a framework for deployment. In this paper, I show the value of implementing a security program like a TQM program, some of the benefits realized by modifying such a program, and how well a security program fits into the TQM model.

Primer:

Like the cost of quality, the cost of security is free in that it pays for itself over time. When you improve a process, the cost of that process has a return on the investment of money, time or both. The cost of improving security returns tangible and intangible benefits. That is, if I improve a process to the point where I am spending less time repairing/rebuilding systems within that process, I can directly measure the timesaving, which translates to money saved in salaries (tangible savings), or it translates to more expendable time on other projects (intangible savings).

Implementing security awareness training programs and best practice guides does wonders for a relatively low cost. To increase our savings, we can limit expenditures by using free tools, utilities and programs. However, to properly evaluate the effectiveness of our plan, we must identify a way to measure how much we are improving. From that point, we can measure the tangible and intangible savings.

Modeling a security implementation plan after the Total Quality Management practices of the 90's seems to be a good fit since security requires continuous improvement over time. One of the primary tools in TQM was the concept of the continuous improvement wheel, also known as the Deming Wheel after the creator Dr. W. Edward Deming. The Improvement Wheel, along with several other TQM concepts, lends itself to improving security quite nicely. This paper is my first attempt

at modifying such a program… and in following the TQM model, it also will be improved over time.

The security program outlined below will eventually cover the following areas as well as identify things that we can use to measure our success or failure in each area. This model should look vaguely familiar to most as components of the SANS Defense-in-depth model.

Network security – route to systems.
System security – route to applications.
Application security – route to data.
Security Awareness – adding to the depth of security.

Over time, the entire TQM practice can be applied to security but for this document, I will only include a few of the major items. TQM is a way of life for a company. It becomes ingrained into the corporate culture. I am not proposing that your company embrace TQM… however, as I stated earlier, it does lend itself very well to the implementation of a security program.

Security is a process, and like any process, it has a definite flow. There are many ways to outline a process and each process can be infinitely complex. For this converstation, I have limited the number and complexity of the processes to keep this paper within a reasonable length. However, the reader should be able to extend this program to any level using the simple techniques defined.

Security Control vs. Security Assurance:

In its infancy, TQM concepts started out as a way to limit the number of errors in a production system. Prior to TQM, a company would set a Quality Control inspector at the end of the production line. To ensure the HIGHEST quality, EVERY unit that came off the assembly line was tested. The problem with this should be obvious… costs rise in both time and money. Quality Assurance removes the inspector at the end of the production line and places the responsibility on each person in the line. Each worker is accountable for the quality going into the unit, and reward systems are implemented based on how the employee reaches the “zero defect” goal.

Relating QC to security, when we find compromised systems after the fact, it’s way too late. By that time, data is stolen or harvested, and storage space is utilized for purposes other than that of the business. Quite honestly, this is the mode I currently find myself in. I am a Security Control manager today. I come across a system that has been compromised and set about repairing that system and scanning for others in the area that may have been touched as well. Because this is such a legacy problem, I find I have very little time left after repairing the system to build a proactive program.

No, we have to build security into our daily routines. In our world, Security Assurance is the only way to successfully ensure a tight ship. We must make sure that EVERY user does their part in producing a secure system. We do this by making them part of the process, after all, they are part of the problem. I believe this

is a different way of looking at security but it is a very important point to get across. Consider it a Holistic approach to security… the CSI approach.

The lack of Security, in today's market, means a lot of things to a lot of people. But one common theme I get from most is fear; fear of loosing data, market value, prestige or time. As a security specialist, we spend most of our time reacting and very little time planning. When we do plan, it tends to be a knee-jerk reaction to a recent event. Dr Joseph Juran, founder of the "Juran Institute"[1], once stated that most of us practice the Ready, Fire, Aim model. That is, we prepare to do something, do it (and spend a lot of money doing it), THEN we look to see if we are hitting the target. Security Assurance means we are going to plan for every aspect of securing for our systems. That implies that we fully understand how our systems are used in the business, who uses those systems and what they need from us to produce a secure environment.

Defined, Security Assurance is a program that works to include every person who uses systems within your domain in the act of securing the whole system. It helps users and technical staff to "Do right things right!"

Metrics:

"Metrics – The branch of prosody dealing with measure and metrical structures."[2]

Before we continue our discussion, let's ponder the following story.

"It has been a year since John implemented the security program. The program cost John an estimated $75k to roll out. He spent the entire wad on software, firewall hardware and awareness training. Tomorrow, He is sitting down with the board of directors to answer a few simple questions… "How is the security program going? How much have we improved over time? How much have we saved in actual dollars?" Now John has to come up with more than just a feeling about how much they have improved the digital security of the firm. He has to come up with actual numbers… and he didn't include that in his plan."

As security professionals, we need to address key issues. Reduction of vulnerabilities, mitigation of risks and improvement of security practices are our goals. However, as stated earlier, it is not enough to have a "feeling" that things are getting better; we must be able to prove it. To show we have improved something we need to measure it before and after we start the program.

CSI is about measurements… it is about knowing, truly knowing how much your process is improving. CSI is a practice that forces you to look into how you do business, and how to track that business with metrics. To gather the information on the business, we use Quality Circles or Focus Teams. These teams really do the work… and what better way to get your message out and to hear back from your end users on what they really need than to include them in the process?

Facilitators:

A facilitator is the security technical expert. This person knows the goals and the risks and is well versed in team building skills. The facilitator is also the one who keeps all the focus teams on target. He gathers data, builds status charts, leads and excites the people involved in the program (which should be everyone).

The facilitator is also a strategic planner. She adjusts the overall program based on the activity of the teams and presses the project forward as needed.

Focus Teams:

Focus Teams (also known as Quality Circles in TQM) are made of a mix of people. In our case, we have technical experts and end users… the toolmakers and the tool users. In large complex companies or campuses, focus teams can provide several key avenues to a successful rollout of a security program. First, they provide a path for you to clearly convey the program details to all levels of the company. Second, they provide feedback on what the end users and administrators need and want from your security program. Third, they provide a pathway for the flow of security awareness issues and ideas between technical staff and end users. Fourth, they enable communication between organizational managers and users.

Focus Teams are tasked with outlining and understanding the flow of business, analyzing the weakness in the processes that handle that flow, selecting metrics to measure, and identifying key result areas that will help us develop attainable goals. They help us clarify which issues are more important than others. For example, they perform "impact changeability analysis". This is one of the techniques that help us get the best bang for the buck. In a nutshell, we select a topic such as passwords and create a question for that topic. Each team member rates what he or she thinks the impact of that topic will be vs. how hard they think it will be to change everyone's thinking. This eliminates the "feeling" or guesswork out of the process. For example, we may have a question such as "Should we Require Password Rotation every 6 months?" I believe that would have a considerable impact on hitting our security goal so I vote for 3. But I also think it would be very difficult to implement in my environment so I vote 1.

Impact Changeability Analysis:
What if we Required Password rotation every 6 months?

Impact	Changeability
1 = Little or no impact	1= Difficult
2 = Some impact	2= Moderate effort
3 = Considerable impact	3= Little or no effort

Priority	Ranking (impact.changeability)
1	(3,3)
2	(3,2)
3	(2,3)
4	(2,2)

Another question might be “Should we require all users pass a simple security test prior to being issued an account?” Here, I believe the impact would be 3 and the implementation would be 3. Comparing the two issues, I would choose the second question (rated at 6) over the first question (rated at 4). Here, we start to see how using this process pairs down and focuses our efforts so we realize that bang for our buck.

In practice, Focus Teams (Quality Circles[3]) should be no more than 6 people in a team. Teams should set schedules for future meetings. Meeting once a month minimum would be sufficient but every other week would be more effective.

Each team must pick a team spokesman. These team leaders will meet once every 1 to 2 months, depending on the team schedules, to compare notes, and discuss KRA’s and PDCA wheel status (outlined below). Team members should be encouraged to disseminate information to their peer groups and administration. This becomes the true security awareness program pathway.

While keeping the model simple for the end user, I suggest facilitators implement a data structure to keep track of the detailed data flow from these groups. Spreadsheets and databases are commonly used during this process. More details on Quality Circles and techniques can be found in “Managing the Total Quality Transformation” by Thomas H. Berry.[4]

A very important note should be made here, the data gathered here will flow into a risk assessment program should you need one.

ID’ing the Process:

As stated earlier, security is a process: a system to produce a product. In this case, our product is a secure system. To evaluate the change in a process, we must first define the flow of that process, i.e. Network traffic flows in two directions. Traffic moves from a workstation to some other point on the network, or from some other point on the network to a given workstation. A good question to ask in the network Quality Circle could be… Should you focus on all traffic at first?

Questions such as these are the starting point of any QC. Once you have outlined a major component of your area of focus, follow that down to the next level. Let’s say you decided to outline inbound traffic flow, what is next? Maybe we should define what good traffic is and what bad traffic is. An example of good traffic is HTTP traffic… or… is that bad traffic? It really depends on what the rule sets are for your organization are. HTTP may be limited for only internal use. Another example might be, inbound DNS requests. They could be a pre-attack scan of your systems… some data mining to better focus the attack. So what is the company or department rule on this? Do you allow DNS requests from outside your domain? I hope you are starting to see what it takes to define a process. You’re tasked to dissect it into molecules and then atoms of information. In this case, an atom of information could be “inbound ICMP-Type8 packets[5].”

Once you have the atoms, how do they fit into the business process? One way of determining this is to implement the Zachman Framework.[6] From Lori DeLooze’s paper “Applying Security to an Enterprise using the Zachman Framework”[7], one can see how modeling the Zachman Framework to your data

architecture will help with defining the flow of information. Below is an example from that paper of how HTTP could be analyzed.

Source	Destination	Source Port	Destination Port	Active Session?	Direction of Packet	Filter Action
Outside	Inside	80	>1023	*	Out	Allow
Inside	Outside	>1023	80	Yes	In	Allow
Inside	Outside	>1023	80	*	Out	Allow
Outside	Inside	80	>1023	Yes	In	Allow
*	*	*	*	*	*	Block

Figure 2: HTTP Router Rules

As you can see, in this example, HTTP data is allowed in most cases. However, your domain may not allow inbound HTTP packets to all systems but only to a specific few. In ID'ing the process, remember we are looking at these general areas of interest, Networks, Systems, Applications, and Security Awareness. Focus teams should be looking into the primary components or enablers of each major area. With Networks, we are looking at Ethernet packets and those devices that enable authorized flow or qualified blocking of those packets. With systems, we focus on system OS and service vulnerabilities, applications on program (code) vulnerabilities and security awareness on the human factor.

You should attempt to define two molecules of a process on your first pass, but you can extend this to whatever level you wish. In addition, two atoms of information should be identified as measurable components related to each molecule. Again, note that there are many factors that could and probably should be included in each of these areas for proper evaluation, but for the sake of brevity, we will keep this model simple.

Start by identifying key factors in each area. Key factors are components such as packet types on a network or ports listening on systems. Since we are identifying key areas in networks, one item I try to track is how many unwanted NetBIOS port probes my systems see on any given day. This data can then be used to build histograms that I can later measure against. One of my objectives is to reduce the number of scans inside my private network... But to know if I am truly reaching my goal, I need to know where I started. Therefore, a baseline is crucial to proving my success and will be crucial to you as well.

Once you have identified the components of your process, you now have to map the logic flow using flow charts. Flow charts help you to visualize the process and by that, see strong points or weaknesses in your network or systems. It will also help you identify problems in management of these processes. Details on how to build flow charts can be found at the following

http://sol.brunel.ac.uk/~jarvis/bola/quality/pfc.html#pfcex1.[8] I also suggest you use Microsoft® VISIO®™ to build with.

Key Result Areas (KRA's):

Key Result Areas provide a system of identifying attainable goals from those things we chose to measure. These are specific areas we are trying to gain the maximum results in. It also allows us to identify those things we DO NOT want to measure… an equally important point when we are looking at saving money. A KRA for security might be to reduce the number of viruses reported in a year by 75%. Another might be to initially reduce the number of vulnerabilities discovered during a random or routine domain wide scan by any degree… but of course, it helps if you have a target so let's say to get the number down by 50% every 6 months.

Now, some things lend themselves to counting such as the number of vulnerabilities located on a system. Other things are not as easy to measure, such as understanding general security practices like how many people in your organization know NOT to follow instructions on an Internet browser pop-up window that says "You have been temporarily disconnected from the network. Please retype your login name and password and hit OK!" To fully evaluate improvements in any system, we must have metrics.

I try to keep it simple when identifying KRA's. I suggest your first time out you limit your KRA's to 3 in each category. This works out to 12 total KRA's. I also suggest you select a sampling period now rather than later. This will help you estimate how long before you should see results. It also helps in explaining to company employees, for example, how often they will be asked questions concerning their knowledge of security. Sampling rates vary depending on the subject mater. Some tools are automatic in nature (such as a NESSUS scanner) and can be sampled far more often giving up a good trend line for evaluation.

Human factors:

I am constantly amazed at the standard unawareness of end users. At first, I thought that they were indifferent to the needs of system administrators in our attempts to provide them with long-periods of stabile systems and services. But I have, of course, come to the realization that it is not indifference but unfamiliarity with computing in general. We often teach end users how to use word processors or spreadsheets to get their jobs done, but little training goes into explaining how the network operates in general. Many users have a difficult time grasping the difference between the C: drive and the Z: drive. To them, they both exist in the same place, on the little box sitting next to their desk. The network cable is just for passing their Email out to the Internet or browsing websites… nothing more. This lack of understanding is a systemic problem, not just a problem with security. But it increases the level of resistance to change. In CSI, resistance to change is a major factor in failure. For us, resistance to change can be directly linked to the level of insecure systems found in just about any system. When was the last time you didn't hear an end-user complaining about having to change their password?

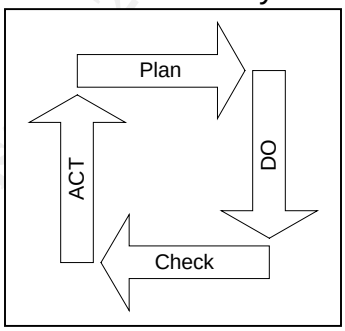
Part of TQM and now CSI is the involvement of end users in the process. CSI is a wonderful enabler for getting some education and enlightenment out to the users who HAVE to use the system we make for them. Of course, this document is targeting just security, but it can be expanded to address any level of "systems operations" within the organization.

Therefore, while you are building your CSI program, make sure to include in you Focus Teams people who are NOT technically inclined. They have a habit of taking gained knowledge back with them and spreading it around… which in our case is a very good thing. Furthermore, when your teams finally come to the process of defining the business, they know the business better than you do by a long shot. When they look at a password aging/complexity policy, you will still get complaints from end users… but they should at least understand why it is important.

That said; don't exclude your technologists either. These people will have to go back to their departments or units and cut time out of their already busy schedules to assist in the implementation of Snort, Nessus and other CSI tools. But being a part of the process will hopefully inspire them to spend more time acting proactively.

PDCA:

The whole point to adding the word "continuous" to this program is to ensure that we keep checking our status. PDCA stands for Plan, Do, Check and Act and is the Continuous Improvement Wheel. In this case, we start by planning to roll out a program, we do this, we check to see if we are making headway and we act on the results of that check. From there, the wheel starts again planning on the action that is needed, doing that action and so forth until we achieve our goals. But we don't stop there… we continue to look for more ways to improve.



In the case of our security plan, this is where the metrics come into play. At first, we are simply gathering data. But when we return to the check portion of our security plan, how far have we gotten? PDCA reminds us that we must continue to evolve our program and our system of checking. Keep that in mind when we set deadlines and schedules.

Tools of the Trade:

There are MANY tools available that perform specific functions we can apply to security. Some of these tools are very expensive but for the most part, we can

accomplish an awful lot with free tools like the following. Since we are talking about security, I have selected a few of the common tools available for use. These tools can be used to gather data for analysis. In addition, tools such as Microsoft's® Visio® 2000 comes with a TQM flow-charting tool as well as others. Excel® also performs many of the charting you will need to analyze your data.

Ethereal – Ethereal is a network-sniffing tool that does a wonderful job of capturing network packets and assisting the user in analyzing the data captured. The latest version, 0.9.8, can be downloaded for free from http://www.ethereal.com/.[9] The tool is easy to use and has been ported to most platforms in use today. Keep in mind that this tool captures network traffic located on the local segment of the system running it. That is, it will only capture data that passes by your system. If your company, department or unit utilizes switched technology, it may be difficult to get a real picture of what is going on in your network. The proper positioning of your ethereal system is important in capturing the true status and health of your network. Elaborate schemes can be implemented to allow capture of "real" network traffic but most of them require access to the routers and core switches. Hopefully, you will have the Network Engineers on one of your Focus Teams. They can help position one or several sniffers across the network to correlate data into a true picture of what is happening.

Nessus – Nessus is an excellent tool for doing some internal scanning. It utilizes the functions provided by NMAP, however, the server only runs on a UNIX/LINUX platform. Once again, the tool is free for download and can be found at http://www.nessus.org[10]. This tool ties quite a lot of capability into one easy to run tool. Nessus is a vulnerability scanner that can be quickly configured to run through a series of tests against a range of IP's. That said, if you spend a little time with it, Nessus can be configured to perform routine scans of your network and store those results for latter evaluation. When utilized in your CSI project, Nessus can provide a path to rapidly assess the current state of the systems while giving you the benchmarks you need. Nessus is at the heart of our security program, locating security problems before they become a risk to the department.

Properly deployed, Nessus opens the door for not only tracking your success, but also ensuring continued security improvement. Since Nessus utilizes programmable scripts for its testing, new vulnerabilities can be (and are) added at a moments notice. Though this scripting language is not for the meek and mild, it is worth learning just so you can test for special events. A very large repository of vulnerability plug-ins is available at the Nessus link here http://cgi.nessus.org/plugins/.

Another powerful feature built into the Nessus server is its ability to allow secure connections from remote clients. The clients have been ported to almost every platform including Windows. With this feature, accounts can be issued to your entire peer IS manager group so they can help in testing the domain. They can securely access the scanner and perform tests when they need to.

PortSentry – PortSentry is a tool used to secure an OpenBSD/LINUX based port blocking tool used to secure a server. It is a perfect mate for your Nessus server. Make sure that when you deploy your Nessus scanner, you spend a lot of

time securing the server. Pick a platform such as LINUX and install PortSentry on that server. PortSentry is free and can be downloaded from http://www.psionic.com/index.html[11]. You do have to register it, but that is a small price to pay. PortSentry utilizes IPTables (more expansive and yet flexible than IPChains) to secure access to the server.

SNORT – Snort is a host based IDS (Intrusion Detection System) that can be deployed in your organization to help you gather some of the detailed data needed in the CSI process. Snort is a rule based system designed to detect intrusion attempts and, with proper configuration and the use of other tools, act upon that attempt. Snort is free and can be configured with MySQL, a relational database similar to Microsoft SQL server. With the use of a SQL database, data can be centrally stored from a distributed series of Snort servers and workstations. This enables you to gather data from a cross-section of your network and later, analyze that data for trends in both security and in planning for future attacks. The data will also help your CSI program and will enable you to judge whether you're hitting your KRA's or not.

For example, let's say you roll out Snort at 5 points across your network. After a few weeks (well, more like a few min's in today's Internet), you notice your network is almost always being scanned from outside systems looking for open, unsecured Windows shares. You now have data on the current trend and define a KRA to drop the scans to zero by years end. A new blocking policy drops outside NetBIOS port scans. After you implement the new policy, you go back to the Snort system and look at the data… it should show a significant reduction in the number of outside NetBIOS port scans. As a matter of fact, there should be NO scans of this nature. Keep in mind that you have not removed the vulnerability; you have limited the threat and this action mitigates the risk.

Once again, using the Quality Circles to disseminate Snort while one central server collects and correlates data is a very good way to both get the information you will need and get group participation into the process.

Snort is also free and can be downloaded from http://www.snort.org[12].

Security Scoring tool – The Center for Internet Security has developed a tool for checking the security on a given server or workstation. The tool checks your system against industry standards and rates your server on a scale from 0 to 10. The objective is to achieve as high a score as possible while keeping in mind that a totally secure system may not be very usable. The Scoring tool is great for testing the relative security of your systems and since it provides a number in its evaluation, this lends itself to the program metrics.

The Scoring tool is free and can be downloaded from http://www.cisecurity.com[13].

Making it happen:

Below is a fictitious scenario based on how the program would progress. In this story, the program has passed the startup period, the data gathering and

planning phases and the team is just now starting to implement solutions into the department.

Step 1 – Getting approval

The SCI program can be implemented at any level in theory. However, the benefits are best realized from a top down implementation. This could mean a unit level, department level or corporate level. In my example, I am implementing this at a department level that has 7 units below that are involved. I have approval and buy-in from the department head. Note a very, very important point here… with buy-in comes funds for expendable resources, hardware and software.

Step 2 – Building the Focus Teams

Given that I have 7 units below me to facilitate, I first looked for technical people involved in the daily maintenance of systems. I have a total of 11 people. I also want to ensure that I have representation from end users so that I can get their needs included into the security plan. I request each unit to locate 2 volunteers from their department. I now have a total of 25 people. I am looking at roughly 4 teams. As a facilitator and subject expert, I will assign each team with a focus area. Each team is tasked with producing between 3 and 6 KRA's as a baseline for metrics.

Team A will gather data on, evaluate and analyze the Network process.

Team B will gather data on, evaluate and analyze the Systems process.

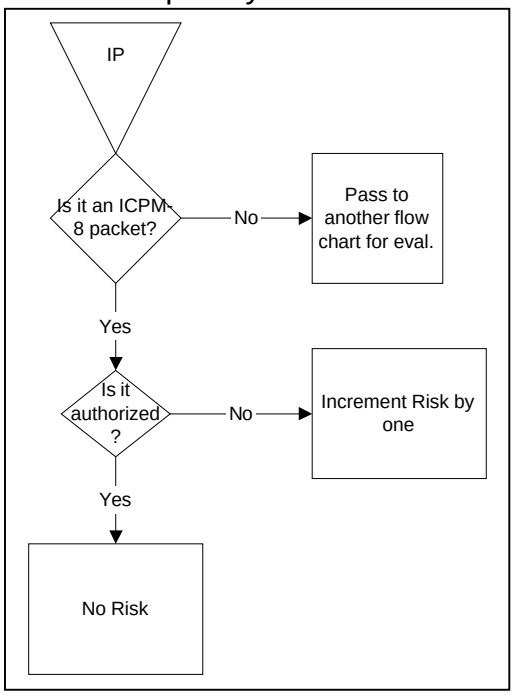
Team C will gather data on, evaluate and analyze the Distributed Applications process.

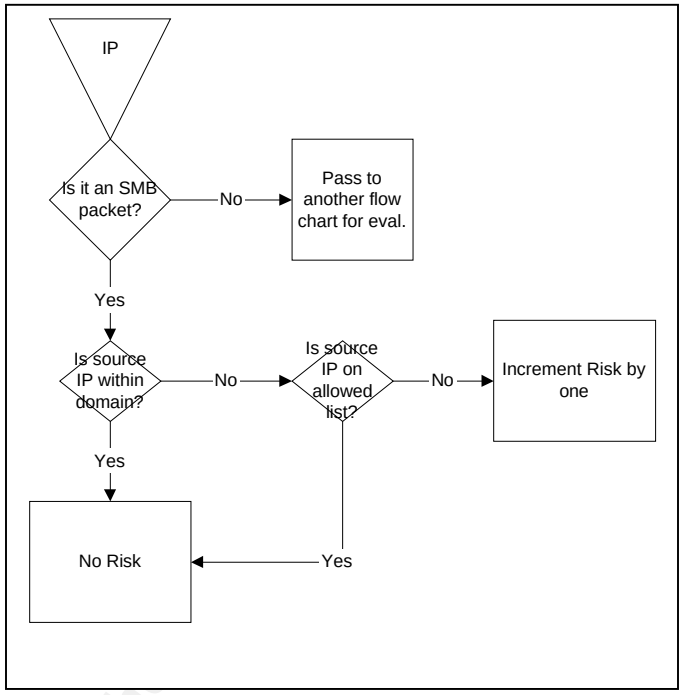
Team D will evaluate Administrative Policies/ Procedures, User awareness and training processes.

Step 3 – ID'ing the processes

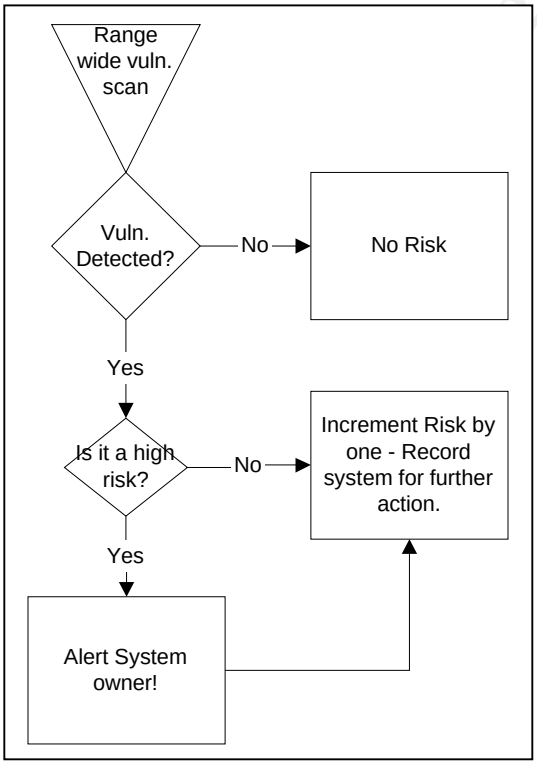
Each team selected a team leader and was assigned an area to focus on. They discussed the processes assigned in their areas. Team A was assigned to Network security. Ideas on metrics ranged from the number of nodes on the network to checking each and every packet that crossed the network. Brief training sessions were established to educate some on what a network REALLY is. The technical people helped those who didn't quite understand it. The end results were that they decided to count the number of routing/switching devices as well as the number of nodes on the network. They also decided to identify aggregate switches, or where the network converged. These devices were then used to gather data for analysis.

Following are a few examples of the initial flow charts use to analyze process flow developed by Team A.





From the flow charts, they were able to identify where they would place limiter devices such as firewalls. It was also evident where they would install sniffers and IDS. In addition, Policies were easier to review and implement based on a clear understanding of how traffic moved into and out of the network.



Team B was assigned to systems. The focus here was more on how each system was administered. Discussions revolved mostly around how each system became insecure. To isolate a starting point for evaluation, the team took the assumption that all systems start off insecure and are made secure by a system admin but become insecure once again over time. I know this is a bit simple but it was a good point to start from. The team identified the fact that user interaction added to increased insecurity by downloading applets from the Internet, or by adding shares without protecting those shares. The team also agreed that it is difficult to ensure that every action taken on a system is monitored. Therefore, the team agreed to work with remote monitoring tools to check for new vulnerabilities. The team also agreed to start proactive work during the

evaluation phase. The flow chart shows how this was accomplished.

Another metric the team noted was to tack the support contact hours per machine. The theory was raised that if the contact hours are low, security vulnerabilities are high. Converse to that, if contact hours are high, security vulnerabilities are low. This is a theory that should prove out in the next PDCA cycle.

Step 4 – Establishing KRA's

Each team defined a couple of KRA's on the first pass. Team A gave us the following.

KRA #1 Reduce total number unlisted/unauthorized network entry points to zero.
KRA #2 Block all unauthorized NetBIOS scans while allowing authorized NetBIOS scans by years end.
KRA #3 Implement a firewall device by years end.

Team B gave us the following KRA's.
KRA #4 Reduce number of "SANs/FBI Top 20" Vuln's discovered by 50% every 6 months.
KRA #5 All users pass simple security awareness test by years end.
KRA #6 Unsecured network shares reduced to zero.

Step 5 – Plan course of action.

Based on the KRA's identified, we were able build a solid implementation plan that targeted key areas of weakness. We started development of a security awareness program to start training our users with. We also located funds for a firewall device. We are currently looking into War-Dialers to scan for modems across the organization.

During the planning phase, we have built management systems to track authority of services across the department. This is expected to help in the future when we go to extend the process. We now know who controls what systems and what part of the network better than we ever did before. We were also forced to audit the network ports in use across our department.

Step 6 – Do what is planned.

Our next course of action is to implement the various plans developed during our discovery phases. This will take a few months to a year. We are still working to schedule hardware installation and establish rules for that hardware. Furthermore, we are working to build better communication channels between the various units within the department. These will be used to alert technical staff in the event that we discover a critical compromise in a system during monitoring.

Several units are developing monitoring systems and sharing their learning discoveries in the process. It was agreed that the monitoring of systems must be

started before the correction of systems begins. This will ensure that we have a routine for cross checking systems after the fact.

Many other events are being planned at this time.

Step 7 – Establish Schedule for Review:

Our current schedule for team meetings is once a month over the next year. At the 6-month mark, we plan a quick review to get a rough sense of where we are in regards to the overall schedule. By the end of the year, we will gather past data, and perform another complete benchmark scan and review of processes. From there, we will check the results of our last years work against the KRA's and start the planning phase once again.

Conclusion:

This structured approach to security will improve over time. The practices borrowed from TQM will give us tools to analyze the complex nature of security and build a strong framework that will ensure continuous improvement. But the most exciting thing about CSI is how it involves end users in the process. It brings them into the program and empowers them to fix and secure their part. CSI cannot be just a buzzword; it must permeate the fabric of the corporation. In practice, it will ensure that when we install a firewall, it will not be the last thing we do… but the first in many actions to secure our part of the Internet.

---

[1] Juran, Joseph. http://www.juran.com/ (Dec 24, 2002)
[2] New College Edition – American Heritage Dictionary of the English Language Houghton Mifflin Co. 1976
[3] Author Unknown. How to implement a Quality Circle. http://www.geocities.com/Heartland/Acres/3257/whatqc1.html (Dec 22, 2002)
[4] Berry, Thomas H. 'Managing the Total Quality Transformation.'Baskerville. McGraw-Hill, 1991, pg 55-74.
[5] Postel, J. RFC792. http://www.faqs.org/rfcs/rfc792.html (Dec 29, 2002)
[6] Hay, David C. THE ZACHMAN FRAMEWORK: AN INTRODUCTION, Essential Strategies, Inc. http://www.tdan.com/i001fe01.htm (Jan 9, 2003).
[7] DeLooze, Lori, September 6th 2001. "Applying Security to an Enterprise using the Zachman Framework" http://www.sans.org/rr/modeling/zachman.php (Dec 29, 2002)
[8] Jarvis, Chris. "Process Flow Charts" from "Business Open Learning Archive", http://sol.brunel.ac.uk/~jarvis/bola/quality/pfcex1, (Jan 11, 2003).
[9] Ethereal Co. home page. http://www.ethereal.com/ (Jan 11, 2003)
[10] Nessus Org. Home Page http://www.nessus.org (Jan 11, 2003)
[11] Psionic Corp. Home Page. http://www.psionic.com/index.html (Jan 11, 2003)
[12] Snort Home Page http://www.snort.org (Jan 11, 2003)
[13] Center for Internet Security Home Page.http://www.cisecurity.com (Jan 11, 2003)

Last Updated: May 28th, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event