Proceedings of Informing Science & IT Education Conference (InSITE) 2015

Cite as: Houghton, R. F. (2015). The creation of the chain-link fence model. *Proceedings of Informing Science & IT Education Conference (InSITE) 2015,* 221-229. Retrieved from http://Proceedings.InformingScience.org/InSITE2015/InSITE15p221-229Houghton1525.pdf

The Creation of the Chain-Link Fence Model

Robert F. Houghton
Idaho State University, Pocatello, Idaho, USA
hougrobe@isu.edu

Abstract

There is much known about logical ways of implementing security measures in an organization. Anecdotal evidence shows that few organizations implement a full suite of measures. This study used a single case to identify problems with a logical implementation. A framework was developed using the literature and these identified problems. The framework was then tested through interviews of experienced professionals.

Keywords: security, security framework, university culture, security creation, policy creation

Introduction

Security of computer systems in educational settings involves a set of well-known activities. Despite the often described activities that should be implemented, many Information Technology (IT) service departments are able to point to simple measures not being implemented. For instance the University of California at Santa Cruz (UCSC) reports that “A faculty physician replied to a phishing email and revealed his or her email username and password. The email account contained private medical information for about 600 patients.” And “Hackers took advantage of a known vulnerability on an unpatched server, potentially putting 30-40,000 student records containing …” http://its.ucsc.edu/security/breaches.html accessed Nov 2014. This paper aims to present an analysis of why it is that people do not implement security measures and a model that identifies essential components of a system to ensure that security measures do become implemented.

Research Questions and Aim of the Study

The study reported here used a longitudinal case study of security in a USA University setting. The aim of the study was to answer the overall research question:

- What are the principal outcomes that must be achieved for efficient implementation of security measures in a University setting by teams of humans?

In order to answer this question an in depth study was made of several stages of the development of a model. These stages answered the questions:

- What problems arise when a logical process is attempted?
- What do these problems and the literature suggest as a model for achieving implementation of security measures in a University environment?
- Does this model seem robust to experienced professionals?

Editor: Eli Cohen

The Case Study

The case involved all of the departments at Utah State University where computing was a significant part of the department workings. The significance was determined by the department employing a person with responsibilities for ensuring the computing facilities were maintained. At the time most departments considered their computing needs to be so unique that they could not use a centralized support service which meant a high level of decentralization of technology support.

Method

The study took place in three phases.

1. In the first phase an attempt was made to identify and implement the most important and pressing security issues at the university. At the end of this process the meeting minutes and outcome documents were analyzed. The analysis identified problem areas with either the identification or the implementation of security processes.
2. In the second phase literature regarding problems of implementation of system processes was analyzed to produce a model that would address the problems from the first phase.
3. The third phase of the study was a validation of the model across the stakeholders at the case study site.

Results

Phase 1

The university employed many individuals who had information technology responsibilities. A task force was formed of members of different departments from around the university and they were assigned the task of creating university-wide security procedures. The task force members will be referred to as “systems administrator (SA).”

The task force needed to find a beginning target for determining the most pressing security issues at Utah State University. To help determine these issues, the task force created a survey in September 2009 to determine how different departments categorize different elements of security. It was hosted on surveymonkey.com and given by invitation to members of the “network managers” email mailing list at the university. This survey was conducted through November 15, 2009 and had 44 respondents. The survey contained a list of security tasks. These security tasks were derived by the task force based upon their experience, brief research into best practices, and information received from peer institutions.

Each respondent was required to categorize each task by required, suggested, or optional. The raw data was then coded by a scoring system of five points for required, three points for suggested, and one point for optional. Each task was then averaged for the total mean score of that procedure. Tasks that scored above four points were initially placed in Tier 1 (Required). Tasks that scored between three points and four points were initially placed in Tier 2 (Suggested), and, tasks that scored below three points were initially placed in Tier 3 (Optional) (see Table 1).

These tasks constituted the computer management procedures for Windows, standard security.

Table 1. ***Task Force Survey Results***

Tier 1 (REQUIRED)	Rating Average
Install/configure anti-virus software	4.95
Configure automatic Windows updates	4.80
Register IP address in OpenIPAM	4.55
Install/configure firewall software	4.37
Update drivers	4.27
Disable the local Windows Guest account	4.12
Disable auto-run	3.39
Tier 2 (SUGGESTED)	
Employ secure user password policies (complexity, length, expiration)	3.94
Convert file system to NTFS	3.91
Uninstall "bloatware"	3.79
Install/configure anti-malware software	3.79
Install/configure SCCM	3.29
Reformat hard drive and install Windows from scratch	3.11
Configure least necessary privileges for the computer owner's account	3.05
Configure automatic third-party software updates, when available (e.g., Adobe Updater)	3.05
Disable all unnecessary services (e.g., utilize Windows Baseline Security Analyzer)	2.94
Rename local Administrator account	2.52
Disable Administrator account	1.85
Tier 3 (OPTIONAL)	
Employ a backup solution (e.g., shadow copy, store-to-network, portable external drive)	2.94
Configure services to use non-default ports (e.g., Remote Desktop)	2.88
Employ security-related group policies via Active Directory (i.e., join Windows domain)	2.82
Install/disallow certain web browsers	2.82
Install security protections specific to installed web browser(s) (e.g., FireFox No Script plug-in)	2.69
Configure power management options	2.68
Employ security-related local group policies	2.60
Install/configure third-party software update notifiers (e.g., Secunia Personal Software Inspector)	2.52

Install Windows from an actively maintained image (e.g., Ghost)	2.50
Install Windows from scratch with slipstreamed service packs and/or patches	2.41
Rename the local Windows Guest account	2.41
Remove local administrator privileges from computer owner's user account	2.40
Configure browser to automatically purge browsing history	2.20
Encrypt the hard drive	1.33

After conducting the survey, the university was given an outside security audit that found some of the tasks should be categorized higher to prevent a greater threat than others. Once the audit results were taken into account, the task force then, with subsequent internal discussion, created the finalized version of the computer management procedures. The task force presented these recommendations to the members of the network managers group and the entire group finalized the computer management procedures.

Once the procedures were approved they fell under the University employee policy. Due to other university policies, there could be no requirement that students use the procedures. However, the procedures were presented as a guideline on how students should set up their computers.

While the task force accomplished its one time job of creating university-wide security procedures, there were other information technology security issues in the university environment that would need to be addresses. Information Technology is an ever-evolving field, and the process of creating procedures needed to be repeated time and time again to address new devices and technologies that arise. As USU grew, the largest growth area in the network was wireless connections. As the internet had increasingly became people’s main source of media, the number of devices that took advantage of this new method of distribution grew. New devices, including the Apple iPad, Motorola Droid, and Google Chromebooks, made it harder to define what an IT device was. Most people would agree that anything computer-related would come under the IT umbrella, like laptops, desktops, networking devices, etc. However, devices like game consoles, blue-ray players, and televisions had wireless internet connectivity built-in.

As more of these devices joined the network, new procedures were needed to define secure methods of using these devices. The process of creating a task force was effective but not efficient. Creating a task force meant that the university used IT professionals outside of their hired duties, created time inefficiencies, and due to the lack of student requirement, did not address the majority of vulnerable computers on the university’s network. It took eight people six months of weekly meetings to create the current procedures in this study. Institutions cannot afford to dedicate numerous resources every time they need a new security procedure. If there was a model to follow for creating security procedures, then anyone who saw a security need could use it and quickly get procedures implemented.

The notes from the task force meetings show the following observations:

1. Although the task force had been set up by the University CIO, most SA were pleased to have been consulted and that the process was partially democratic.
2. The SA initially came together with the view that their own opinions would hold sway. For example one head postponed his vacation for six months to be able to provide his input. As the task force proceeded it became clear that the outcome was to be for a centralized set of processes and at this point resistance began.
3. In a decentralized system each unit head has a worldview that their responsibility is to

their particular user base. This meant they were not prepared to accept a control above them that would interfere with them answering the specific problems of their user group. Each head on the task force was more concerned about the specific departmental needs than the security needs of the organization as a whole. The advantages postulated by the proposed system activities would bring benefits but remove power from the head and potentially decrease perceived value in the department. San example is one head who did not want any other members of the university IT community to have any type of access on his system. As a result, the person did not want to join any centralized administrative process, specifically the university's global Windows domain.

4. Some of the initial resistance to implementing centralized processes was that the SA did not always understand a new tool or procedure. The reaction of dismissal in the face of ignorance may been a result of SA having to be completely independent up until this time and so needing to see themselves as "all knowing." As new tools were explained and the SA realized the efficiencies to be gained from them they quickly began to become supportive of the new processes. For example one head did not know any of the benefits of using organizational units (OU) within Windows Group Policy. Once this concept was explained, he realized that he could accomplish more security goals with less effort when he was working as part of the group.

Phase 2: Creating the Model

The available literature showed no single standard for creating security policy or procedures. There were helpful guidelines provided by the National Institute of Standards and Technology (NIST) that contained references to computer security, the International Information Systems Security Certification Consortium, Inc., $(ISC)^2$ maintained a global standard for information technology security professionals, and Microsoft offered a class on how to properly configure Windows in various environments.

These resources, however, did not contain any method of a higher level framework to create security procedures. They had either very specific guidelines for a subset of requirements (e.g., Microsoft's configuration for small business and $(ISC)^2$ model for penetration testing), or in contrast, they were overly broad (like NIST's statement of security management and assurance). In part, NIST stated, "Ultimately, responsibility for the success of an organization lies with its senior management. They establish the organization's computer security program and its overall program goals" (NIST, 2012). A search of the academic research topics for guidelines, frameworks, procedures, and other keywords was used to help develop a framework so that the efforts of the task force could be replicated more efficiently.

A major paper influencing the development of the Chain-Link Fence Model was by Da Veiga and Eloff (2007). Their paper had a framework for IS governance, rather than security directly. Their framework suggested changes to the organization's security culture. It discussed the need for management buy-in to change the culture.

Buy-in

Backhouse et al. (2006) addressed power. Without full buy-in from those in power, the process could not be completed. This and the Da Veiga and Elorff (2007) article showed how the process starts with buy-in. In their study of spyware Warkentin, Xin, and Templeton (2005) found that understanding the problem, or buy-in, would help in the use of anti-spyware software. Gillespie (2009) stated that buy-in was necessary for enhanced security.

Ease-of-Use

An influence of the Ease-of-Use component is a standard of IS research in the Technology Acceptance Model (TAM) by Davis (1989). His research into TAM suggested that ease-of-use helped users develop the necessary skills to adopt new technology. In the current study, the author predicted that with new security procedures, there would be new-to-users technology that they would need to accept. Mustonen-Ollila and Lyytien's study of IS history (2003) also showed that one of the most influential factors in progressing IS process was ease-of-use. In 2009, Scholz discussed basic mistakes that were caused by overly confusing the security process. This confusion suggested that ease-of-use must be part of the process in order to prevent human error.

Implementation

At this point, the model's framework showed a logic gap. To go from organizational buy-in to ease-of-use, users needed the actual procedures to be written with full documentation. Bresz (2004) researched how hospital information technology dealt with Health Insurance Portability and Accountability Act (HIPAA). One of the key parts of HIPAA was full documentation before changes could be made to a system (i.e., a procedure for implementation).

Von Solms and von Solms's (2004) article of the 10 sins of IS security suggested that training both the culture and users were very important to maintaining proper security. Vroom and von Sols (2004) stated that up to 48 percent of breaches were accidental in nature. This showed that proper implementation of procedures could greatly reduce security issues. This led to the inclusion of implementation as a component of the Chain-Link Fence Model. Sasse et al. (2001) showed that users needed training to avoid being the weakest link in the security process.

Effectiveness

The buy-in component was necessary for development, and the ease-of-use component was necessary for users to change the overall culture of security. Rees et al. (2003) created short-term security policies. The usefulness of these policies was determined by feedback from stakeholders. This led to the inclusion of evaluation as part of the Chain-Link Fence Model framework. Vroom and von Solms (2004) showed how auditing was an effective tool for gauging how an organization complied with their security policies. Furnell et al. (2000) specifically discussed how effectiveness could also lead to adoption of security measures, showing how the starting point of a future model could be at the end of a process.

As the components were identified, the model of the framework needed to be expounded upon. The first diagram for the model of the framework was a straight line (see Figure 2).

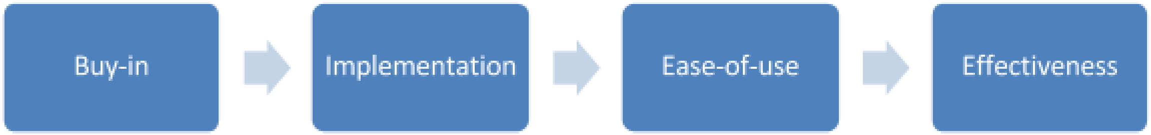

Figure 2. Developmental Step 1 of Chain-Link Fence Model.

The straight line, however, implied that once the process was complete, the goal was achieved. As the Rees et al. (2003) article stated, there needed to be a constant feedback part to the model, in essence closing the loop. This led to a different model demonstrating an ever-completing cycle required by constant feedback (see Figure 3).

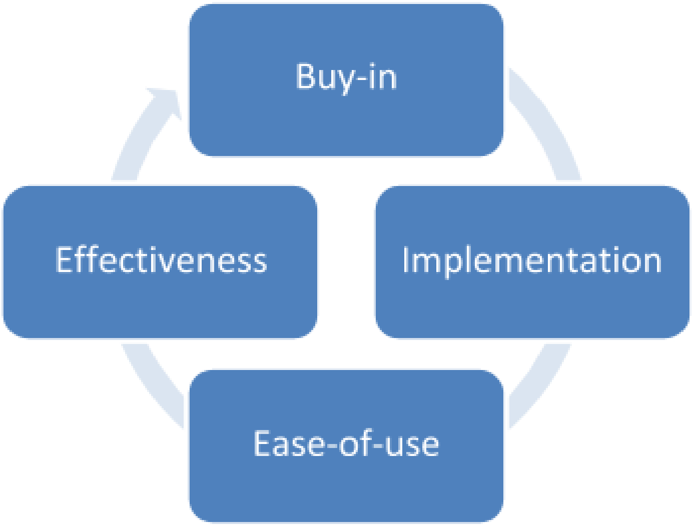

Figure 3. Developmental Step 2 of Chain-Link Fence Model.

Further thought led to the realization that each component was not a stand-alone component. Each component is a precursor to the next. Each component relies on the next to influence the component opposite it. They needed interaction to support each other. Lastly, evidence was not found to show interaction between the opposite components. Also, while the development of CLFM did start with Buy-in, the circular model could start from any component and can only run forward in sequence.

This led to back the final completed framework and the name for the Chain-Link Fence Model as shown in Figure 4.

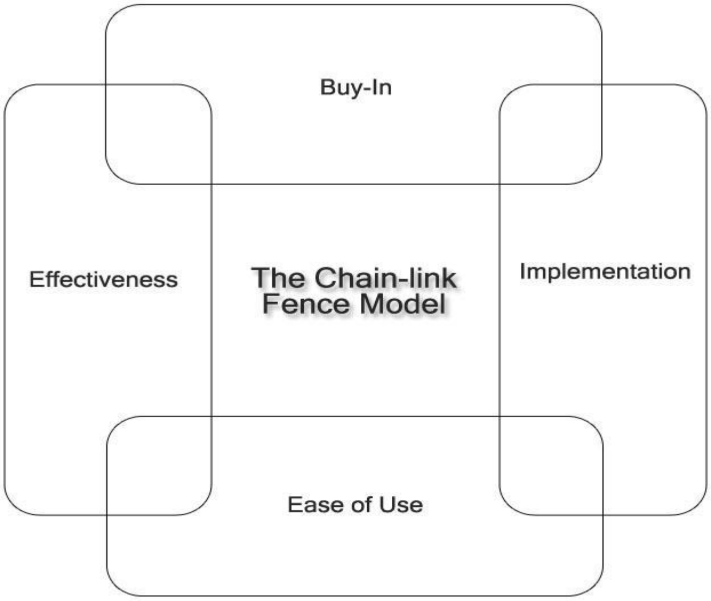

Figure 4. The Chain-Link Fence Model.

Phase 3: The Chain Link Fence Model (CLFM) was the Final Outcome of the Study.

Conclusion

Security in the open and complex environment of a University is a difficult problem. Systems include both those of high sensitivity such as finance and student records and open access such as allowing students unfettered access to the internet.

This case study showed that a University IT resource includes many stakeholders with little commitment to institution wide policy and future considerations. When asked to rate the importance of a range of tasks the task force surprisingly found that encryption, backup, least privilege requirements, and other staple security issues were ranked lower than items that the Windows operating systems provide automatically. An analysis of notes of meetings of the task force showed that an institutional stance was required since the individuals had different and lesser knowledge of current technology then their job responsibilities would normally require.

This model emphasizes the interaction between technology, management, and user outcomes. Each component of the Chain-Link Fence Model (CLFM) focuses upon solving a different and unique problem but creates the overall goal of developing security procedures.

The CLFM model provides a sequentially linked set of precursor outcomes that must be achieved if security is to be implemented. This model concentrates on the humans who must carry out the implementation. The literature is replete with examples where the failure to successfully involve humans has led to failure. This model can help reduce that source of failure

The next work to be done is to evaluate the CLFM in an implementation of new security features. It may also be possible that this model, developed in a security environment, may be applicable to other systems implementations.

References

Backhouse, J., Hsu, C., & Silva, L. (2006). Circuits of power in creating de jure standards. *MIS Quarterly*, *30*, 413-438.

Bresz, F. P. (2004). People–Often the weakest link in security, but one of the best places to start. *Journal of Health Care Compliance*, 6(4), 57-60.

Da Veiga, A. A., & Eloff, J. P. (2007). An information security governance framework. *Information Systems Management*, *24*(4), 361-372. doi:10.1080/10580530701586136

Davis, F. D. (1989), Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, *13*(3), 319–340.

Furnell, S. M., Dowland, P. S., Illingworth, H. M., & Reynolds, P. L. (2000). Authentication and supervision: A survey of user attitudes. *Computers & Security*, *19*(6), 529-539.

Gillespie, M. (2009). Untitled. *Computer Weekly*, 106.

Mustonen-Ollila, E., & Lyytinen, K. (2003). Why organizations adopt information system process innovations: A longitudinal study using diffusion of innovation theory. *Information Systems Journal*, *13*(3):275–297.

NIST, Security Management & Assurance. (2012). Retrieved from http://csrc.nist.gov/groups/SMA/index.html

Rees, J., Bandyopadhyay, S., & Spafford, E. H. (2003). PFIRES: A policy framework for information security. *Communications of the ACM*, *46*(7), 101-106. Retrieved from EBSCOhost.

Sasse, M. A., Brostoff, S., Weirich, D. (2001). Transforming the "weakest link": A human-computer interaction approach to usable and effective security. *BT Technology Journal*, *19*(3), 122-131.

Scholz, J. A. (2009). Securing Critical IT Infrastructure. *Information Security Journal: A Global Perspective*, *18*(1), 33-39. doi:10.1080/19393550802644640

Von Solms, B., & von Solms, R. (2004). The 10 deadly sins of information security management. *Computers & Security*, *23*(5), 371-376.

Vroom, C., & von Solms, R. (2004). Towards information security behavioural compliance. *Computers & Security*, *23*(3), 191–198.

Warkentin, M., Xin, L., & Templeton, G. F. (2005). A Framework for Spyware Assessment. *Communications of the ACM*, *48*(8), 79-84.

Biography

Robert F. Houghton has specialized in the hardware aspect of Informatics. His research focuses upon informatics security with human-computer integration. His background in informatics security provides a large knowledge base for sharing experiences with HIPAA, FERPA, and PCI data security standards. Dr. Houghton joined Idaho State University in Spring of 2014