Cyber Security Training: ANN as an Evaluator and Informant

Liang Kong, Michael Haney, Carolina L and David G

Abstract— It is often agreed that security training and education is very important to the risk management strategy of an organization. However, it is not often possible to quantitatively measure the effectiveness of the delivered training. This paper presents a technique to help evaluate security training effectiveness through automated means. This technique employs artificial neural network as an informant to help analyze data. A case study on p2p downloading is presented.

Index Terms—**Training, Artificial Neural Network, Network Monitoring**

I. Introduction

Protecting cyber infrastructure of organizations across the world is vital. As technology improves every day, human factors are left as the primary weak link. A properly educated and trained workforce is indispensable in protecting an organization against a variety of cyber attacks (Saltzer & Schroeder, 1975). Cyber security training has become a necessity in today's virtual world.

However, simply implementing a cyber security training program (Dodge, Ragsdale, & Reynolds, 2003) does not guarantee its effectiveness. As with all training programs, it is imperative to determine the initial training needs, develop the training program based on the identified needs, implement the training, and evaluate the effects of the training. Unfortunately, evaluation is rarely conducted. Despite the availability of training options, it is difficult to determine whether one option has the intended effect or is more effective than another option. Without evaluating the programs that have been implemented, organizations can not prioritize resources towards a preferred course of action. Thus, it is necessary to evaluate the cyber security training program that an organization elects to employ. This paper proposes a novel technique for determining the effectiveness of cyber security training programs by processing security data with an artificial intelligence algorithm to determine policy compliance.

We discuss several key background concepts that support understanding the focus and results of this work in the next section. In Section III, we discuss our methodology and the reference model for our neural network testing with a p2p downloading case study. In Section IV, we present the results of our testing. We then conclude with a discussion of the findings and potential for future work.

II. Background

A. Security Training Evaluation

As mentioned earlier, organizations do not typically evaluate cyber security training. In order to determine whether a cyber security training solution has actually been effective, the organization should adopt the Integrated Model of Training Evaluation and Effectiveness (IMTEE) (Alvarez, Salas, & Garofano, 2004), which consolidates over ten years of training evaluation research and evaluation models into a single methodology.

This model consists of four levels of assessment that provide a holistic evaluation of a particular training program. The first level is a needs analysis, in which the organization, the individuals, and the individuals' tasks are evaluated to determine if and what training is necessary (Noe, Hollenbeck, Gerhart, & Wright, 2004) before resources are expended (Cascio & Aguinis, 1998). The needs analysis results are then used to develop the training program and materials. The second and third levels of the IMTEE are designed to assess training content and design, changes in learners, and organizational payoffs. The fourth and final level of the IMTEE assesses training effectiveness variables. This model prescribes specific methods for evaluating these areas.

While utilizing this training evaluation model can benefit many organizations that use cyber security training programs, it is not a new idea. This type of evaluation has been used for several years to determine the effectiveness of different types of training programs. In this paper, we present a novel

Acknowledgments: This material is based on research sponsored by DARPA under agreement number FA8750-09-1-0208. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, or DARPA or the U.S. Government.

L. Kong is with the University of Tulsa, Tulsa, OK 74135 USA (e-mail: liang-kong@ utulsa.edu).

M. Haney is with the University of Tulsa, Tulsa, OK 74135 USA (e-mail: micheal-haney@ utulsa.edu).

C. Lindemuth is with the University of Tulsa, Tulsa, OK 74135 USA (e-mail: caroline-lindmuth@ utulsa.edu).

D. Greer is with the University of Tulsa, Tulsa, OK 74135 USA (e-mail: david-greer@utulsa.edu).

technique to evaluate security training effectiveness with an artificial neural network.

B. Security Policy Metrics and Monitoring

When providing security training to members of an organization, the desired outcome is some improved behavior of the users with regards to policy compliance. There are several potential technological means for measuring a user's behavior on a system. One example of an extreme measure might be to install a keystroke logger on every system with a keyboard, a process monitoring agent on every system with a network connection, and cameras on each user's desk to confirm the identity of the person at the keyboard. Some means of behavior detection from these sources of data might be screenshots of the user's monitor compared with records of sites visited, search terms used, the contents of forms uploaded to websites, and whether the user appeared "nervous" or "guilty" based on physiological characteristics observable in the camera images (presumably by a trained expert).

System and application log analysis can often provide good indicators of user behavior at much lower expense. Systems may log which processes are running and for how long, as well as which files were accessed, updated, or deleted during a user session. While this data is available in many cases by built-in operating system tools or installed agents and can be centrally stored and queried, the large volume of data available makes for a quickly diminishing return on investment.

Network security monitoring (Bejtleich, 2013) describes network security monitoring as consisting of several components: system log review, network intrusion detection systems (IDS), using both signature-based and protocol or application anomaly reporting, deep packet inspection (i.e. the review of network payloads and communications content), and traffic analysis. The advantage of these monitoring techniques is that they can be done passively and so do not require installation of a system agent on every user's computer.

With signature-based IDS, network communications can be reviewed to match specific patterns of content for undesired elements. This is normally done for determining intrusion attempts, but may also be used to detect certain user behavior. Egress filtering is the process of reviewing all network traffic outbound from an organization and is usually focused on user activities (Shabtai, Elovici, & Rokach, 2012). Deep packet inspection and content filtering are terms used to describe the review of application-layer user content and not just "header" information that describes which Internet site a user connected to.

Application proxy logs may also be used to capture and review user content. Secure Socket Layer (SSL) provides end-to-end encryption of user communications, but some proxies will provide a point for SSL decryption and recording of communications (Jarmoc, 2012). Each of these network security monitoring techniques provides progressively more detail for measuring user behavior, but also progressively more potential for invasion of privacy of user communications.

In this study, network monitoring is used to help tracking user performance on p2p downloading.

C. Netflow Data

A less invasive technique of user monitoring is traffic analysis. Traffic analysis consists of reviewing the types of network applications and patterns of communication that may indicate user behavior without the need for content inspection. Traffic analysis may also work in the presence of network encryption such as SSL because the specific payloads do not need to be reviewed, only the pattern of which computer addresses are communicating with which others. User identities may also be protected with traffic analysis if Internet Protocol (IP) addresses are reviewed without tracking which address corresponds to which user (which is possible in environments with dynamically assigned IP addresses).

In this research, in order to balance the effectiveness of monitoring user behavior and the privacy concerns of users, we perform traffic analysis by passively recording network data. The network data is collected from an institute following Cisco netflow specifications (Claise, 2004). It contains source IP address, destination IP address, protocol, starting time, duration, packet count, bytes and flows for all network traffic.

D. Peer-to-Peer Networking

Peer-to-peer networking (p2p) is a class of applications that work by distributing communications among user systems as "nodes", rather than using the more traditional model of individual client-server communications. This type of communication helps to eliminate bandwidth bottlenecks in network communications; in the same way, it is used to avoid "choke points" in networks that may be used for control or content filtering. Often, though not always, p2p networking is used to hide the nature of network communications and is thus used for illegal sharing of files with copyright protected materials (e.g. mp3 songs, dvd movies, or "pirated" software). Many instances of viruses or malicious Trojan Horse programs have spread through p2p networks as well (Kalafut, Acharya & Gupta, 2006). Thus, this is one type of negative user behavior that is often cited in organization security policy, and may be targeted by a security training program.

Though there are many different applications which utilize p2p networking, one of the most common and popular for file sharing used on networks today is the Bittorrent protocol (Pouwelse, Garbacki, Epema, & Sips, 2005). Because of its common use and the presence of specific indicators making it relatively easy to identify, this protocol is used as one of the key targets of our monitoring efforts.

E. Artificial Neural Network

An artificial neural network (ANN) is inspired by models of brain and behavior (Arbib, 1987). ANN has been used in many disciplines where an intelligent system is needed (Simpson, 1990). Since ANN is a naïve simulation of brain function, it has a learning ability which allows it to be trained to solve problems in different domains. It also has tolerance of fault input and localized memory which allows it to react correctly to a wide spectrum of inputs (Sequin & Clay, 1990). ANN is composed of elements that perform in a manner that is analogous to the most elementary functions of the biological neuron (Wasserman, 1989). These elements are called *perceptrons*, which are organized in layers and are interconnected. Information flows into ANN from an input layer and travels via unidirectional channels through hidden layers to an output layer.

ANN can be trained to complete complex analyses and provide detailed information about an organization's cyber security standing. In our research, ANN is applied to evaluate network security and a p2p downloading prevention measure in an institute setting. We hypothesized that the ANN created will allow us to evaluate the impact of the current p2p downloading prevention measures, as well as identify network users as those who are more likely to be engaging in p2p downloading.

III. Methodology and reference implementation

A. *Problem Statement*

In this study we aim to help an institute with p2p downloading problems. This institute receives on average 100 DMCA volation emails (Piatek, Kohno, & Krishnamurthy, 2008) per year. Those emails are violation notifications when a user is detected downloading and sharing copyright materials via p2p networks.

This institute tries to mitigate p2p downloading by sending out warning emails to general users. It is important for the institute to send out those emails at the right time and deliver to the right audience in order to optimize effectiveness. In this case, effectiveness can be determined by a measurable drop in p2p file sharing activities.

B. *Artificial Neural Network Structure*

In this study, we set up an artificial neural network containing an input layer of five input nodes and an output layer composed of two output nodes as shown in Figure 1 with pyBrain (Schaul, et al., 2010). The model had no hidden layers between the input and output layers which helps ANN converge quicker and works for linearly separable input.

The five input nodes included total flow, total packets, bytes per second, packets per second, and bytes per packet. Due to the nature of input data, every dataset is a mixture of normal network traffic and p2p downloading traffic. There is not an easy way to separate two kinds of traffic from netflow data. This data is provided by switches and routers that merely record aspects of each network communication flow that is seen passing a given network interface. As such, all types and categories of network communicaton are recorded with equal weight. Therefore in this implementation, two output nodes were set with respect to two classification classes (normal and p2p traffic).

There are two stages in developing an ANN (White, 1989). They are the supervised learning stage and the classification stage. In the supervised learning stage, the network receives examples with expected results and tries to adjust itself to yield the same results. With enough training iterations, the network will converge such that more training will not improve the performance. After the network has converged, it is ready to accept inputs and make decisions and classifications based on the training received.

C. *Netflow Data Collection*

Using netflow data output from institute network switches sent for collection on a research server, data were gathered on all institute network users for three weeks. We received an average of 3 gigabytes of data each day that data was collected. Data that was collected from the first week was used to train the ANN program and was not included in the reported data analysis. During the first week of official data collection, data was obtained for network users to establish baseline levels of p2p-downloading likelihood prior to the institute intervention. The intervention consisted of an e-mail from an institute official reminding the student body of the sanctions for p2p downloading. The e-mail was distributed to the entire institute. Following the e-mail, data was collected on network traffic for an additional week. Data was not linked to individual users but was identified according to a unique IP address which is assigned to a user dynamically when one connects a system to the network. Data was collected on a total of 5,360 IP addresses.

To analyze this data, we took a statistical approach which aggregates data after every 24-hour period. For each time period, we examined total flow, total packets, bytes per second, packets per second and bytes per packet.

D. *Data Preprocessing*

Before the data was entered into the artificial neural network, it had to be normalized into a 0 to 1 range. Two steps were taken to normalize and re-curve the data. First, we divided raw data by the daily maximum. For example, if the highest single user uploaded 70G and User 1 had a total upload of 7G, User 1 would have a bytes value of 0.1. Since the daily maximum can be high, the first step forces data to fall into a narrow range of small values. To make the value differences more meaningful, a non-linear function was used as a second step:

$$f(x) = (e^{x}-e^{-x})/(e^{x}+e^{-x})$$

E. *Arranging Data for Training*

To train the artificial neural network for classification purposes, we needed examples from each category as learning material. We classified users to normal users and suspicious p2p downloaders.

Given the difficulty of manually collecting enough p2p downloader examples by observing network flows, we used a list of Bittorrent protocol trackers as reference. Those who constantly communicated with the trackers were considered to be p2p downloaders, and their data were used as examples for suspicious p2p downloaders. Bittorrent is only a part of all p2p traffic streams from various applications, but most p2p traffic streams share similar characteristics (Madhukar & Williamson, 2006).

Once a user is determined to be using Bittorrent, the netflow data of this user of this particular day is used as one example. On average, we obtained 45 examples per day out of more than 1000 users. Other users were generally considered to be normal users. To match the number of suspicious examples, we randomly chose 50 users per day to be used as normal user examples.

F. *Training the Artificial Neural Network*

In this study we used nfdump (Haag, 2005) for data collection and data-mining. Python was used as the programming language to process and arrange data. Pybrain is an artificial neural network package in Python and was used to construct and train our network. In this network, the 5 input nodes took numerical inputs between 0 and 1. Output nodes yielded a value pair [a, b], where “a” represents the likelihood of being normal user and “b” represents the likelihood of being a suspicious p2p user.

In the training stage, we applied supervised learning to the constructed network. Normal users’ examples were trained with the expected result [1,0], while suspicious users’ examples were trained with an expected result of [0,1]. We used one week of data for this training. We trained 500 iterations on each day and repeated the process 10 times.

G. *Data Analysis*

ANN produced a value for likelihood of p2p downloading for each IP address, which summarized network activity for a single day. P2p downloading likelihood values were then calculated where a 0 value corresponded to no likelihood of p2p downloading and 1 corresponded to a 100% likelihood of p2p downloading. In order to analyze average network usage, likelihood averages were computed for both weeks of data collection for every single day after the first week, resulting in a weekly average prior to and following the institute prevention e-mail. These standardized weekly averages were then used as the basis for data analysis. In order to examine types of network users, all IP addresses were ranked according to the first week’s likelihood average. The IP addresses falling in the highest 5% according to likelihood were labeled “high likelihood” cases, while the remaining 95% were labeled “low likelihood.” To examine differences between high and low likelihood groups, a “change over time” variable was computed by subtracting the week 1 average from the week 2 average. Thus, negative “change over time” values represent a decrease in likelihood, while positive “change over time” values indicate an increase in likelihood following the e-mail. All statistical data analysis was conducted using SPSS statistical software.

IV. RESULTS

A. *Artificial Neural Network Accuracy*

In this research, the artificial neural network is trained with an incomplete data set (Bittorrent downloading), and used against general p2p downloading. Thus it is difficult to use a traditional way to evaluate the effectiveness of this network.

Instead, we focused on the top 50 suspicious users, and tested every single one of them. Of these, 35 had constant communication streams with Bittorrent trackers. All of them match the profile of a p2p downloader, from which we were able to extract the following common features:

1. Relatively large volume of uploading traffic,
2. Relatively many concurrent connections to different destinations
3. Relatively long total transmitting time per flow.

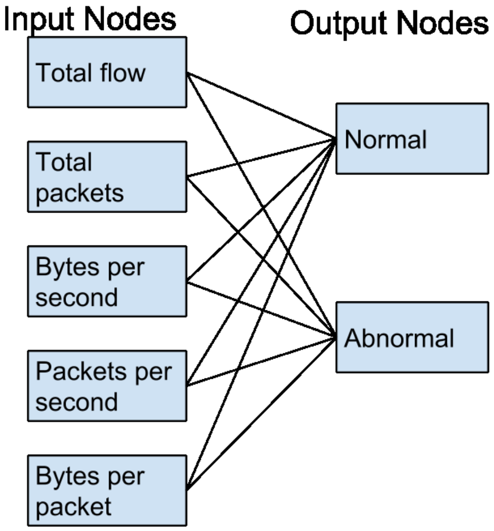
Thus it is safe to say this artificial neural network is capable of helping identity p2p downloaders in this given network. By extracting the above data elements from each stream and comparing them to data regarding the whole network, grouped by IP address, those with significant deviations from the mean can be identified as high-liklihood p2p users.

B. *Training Evaluation*

In this section we are presenting the ability of ANN reflecting user’s behavior changes. We are examining two groups (high likelihood and low likelihood) of users during time before and after the institute sent out a training email. This allows measurement of whether the intended effect is observed to a greater degree in the high-liklihood user group.

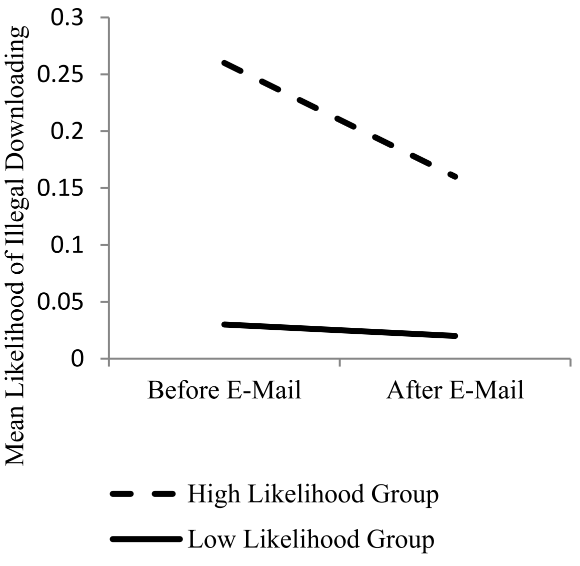
In order to determine whether the institute prevention e-mail significantly decreased likelihood of p2p downloading from baseline, a paired-samples t-test was conducted with significance set at $p < .05$. Results indicated that p2p downloading likelihood before and after the e-mail were

Fig. 1 Artificial Neural Network Structure.



significantly different: t(5359) = 33.38, p = .000. Mean baseline likelihood was .04 (SD = .07) and mean likelihood after the e-mail was .02 (SD = .05). Therefore, likelihood of p2p downloading decreased measurably from prior levels after the e-mail was sent.

Figure 2. Mean Likelihood Before and After Prevention E-mail by Likelihood Group



To determine whether pre- to post-e-mail changes in likelihood were significantly different between the high and low likelihood groups, an independent samples t-test was conducted with significance set at p < .05. Results indicated that change over time significantly differed between groups: t(289) = -17.72, p = .000. The high likelihood group demonstrated a larger decrease from pre- to post-e-mail likelihood (M = -.10, SD = .08) than the low likelihood group (M = -.01, SD = .03). Therefore, the high likelihood group demonstrated a larger decrease in likelihood of p2p downloading following the intervention e-mail than the low likelihood group (See Figure 2).

V. Conclusion

A. Discussion of Results

This research demonstrates that ANN has the potential to be a useful tool in designing, implementing, and evaluating cyber security training programs. First, because ANN was able to provide information through which the e-mail effectiveness could be evaluated, it can be used as a tool to determine to the usefulness of an organization's current and/or proposed training programs. Results revealed that the current prevention e-mail resulted in statistically significant decreases in p2p downloading behavior. Additionally, because ANN was able to make a distinction between high and low likelihood p2p downloaders, it can be a useful tool to pinpoint the employees of a company who are in need of cyber security training. This is particularly important since training can be very costly financially and requires employee time that could be spent on job-specific activities.

In addition to distinguishing between types of employees, this research demonstrates that ANN can provide information on differential reactions among groups to various types of training programs. Specifically, results demonstrated that high likelihood p2p downloaders reacted to the e-mail with larger decreases in p2p downloading behaviors than those users displaying lower likelihood behaviors.

B. Future Work

Although this study only examined one type of training, future research could explore various training methods to determine if different types of users are more responsive to one kind of training over another. One limitation of the current study is the short-term nature of the observations. Although we were able to examine the short-term training effectiveness, future studies could look at change over longer periods of time to assess how long training effects last, which would provide practitioners with information on how often training should be re-administered.

These suggestions are only a starting point for implementing ANN as a training tool and evaluator. ANN opens up new doors by allowing quick and efficient data analysis, which has never before been feasible by solely utilizing human capacities. As cyber security becomes increasingly important in the business world, it will be critical to implement training programs in the most effective and cost-efficient manner possible. ANN has the potential to be an invaluable tool to ensure that the most appropriate training is administered to employees who need it and that the training chosen results in measurable reductions in behaviors that threaten an organization's cyber security.

VI. References

Alvarez, K., Salas, E., & Garofano, C. M. (2004). An integrated model of training evaluation and effectiveness. *Human resource development Review, 3*(4), 385-416.

Arbib, M. A. (1987). *Brains, machines, and mathematics* (Vol. 197). Springer.

Bejtlich, R. (2013). The Practice of Network Security Monitoring: Understanding Incident Detection and Response. No Starch Press.

Cascio, W. F., & Aguinis, H. (1998). Applied psychology in human resource management.

Claise, B. (2004). Cisco systems NetFlow services export version 9.

Dodge, R., Ragsdale, D. J., & Reynolds, C. (2003). Organization and training of a cyber security team. *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, *5*, pp. 4311-4316.

Jarmoc, J., Unit, D. S. C. T., & Intelligence, T. (2012). SSL/TLS interception proxies and transitive trust. *Black Hat Europe*.

Haag, P. (2005). Watch your Flows with NfSen and NFDUMP. *50th RIPE Meeting.*

Kalafut, A., Acharya, A., & Gupta, M. (2006, October). A study of malware in peer-to-peer networks. In Proceedings of the 6th ACM SIGCOMM conference on Internet measurement (pp. 327-332). ACM.

Kumaraguru, P., Cranshaw, J., Acquisti, A., Cranor, L., Hong, J., Blair, M. A., & Pham, T. (2009). School of phish: a real-world evaluation of anti-phishing training. *Proceedings of the 5th Symposium on Usable Privacy and Security*, (p. 3).

Madhukar, A., & Williamson, C. (2006, Sept). A Longitudinal Study of P2P Traffic Classification. *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2006. MASCOTS 2006. 14th IEEE International Symposium on*, (pp. 179-188).

Noe, R. A., Hollenbeck, J. R., Gerhart, B., & Wright, P. M. (2004). *Fundamentals of human resource management* (Vol. 2). McGraw-Hill.

Pouwelse, J., Garbacki, P., Epema, D., & Sips, H. (2005). The bittorrent p2p file-sharing system: Measurements and analysis. In Peer-to-Peer Systems IV (pp. 205-216). Springer Berlin Heidelberg.

Saltzer, J. H., & Schroeder, M. D. (1975). The protection of information in computer systems. *Proceedings of the IEEE, 63*(9), 1278-1308.

Schaul, T., Bayer, J., Wierstra, D., Sun, Y., Felder, M., Sehnke, F., . . . Schmidhuber, J. (2010). PyBrain. *The Journal of Machine Learning Research, 11*, 743-746.

Sequin, C. H., & Clay, R. D. (1990). Fault tolerance in artificial neural networks. *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*, (pp. 703-708).

Shabtai, A., Elovici, Y., & Rokach, L. (2012). A survey of data leakage detection and prevention solutions. Springer.

Simpson, P. K. (1990). Artificial neural system—foundation, paradigm, application and implementations. *Artificial neural system—foundation, paradigm, application and implementations*. Pergamon Press, New York.

Wasserman, P. D. (1989). *Neural computing: theory and practice.* Van Nostrand Reinhold Co.

White, H. (1989). Learning in artificial neural networks: A statistical perspective. *Neural computation, 1*(4), 425-464.