# A Study of Security Administration Errors

Kartik Markandan[*]

Laboratory for Education and Research in Secure Systems Engineering
lersse.ece.ubc.ca
University of British Columbia
Vancouver, Canada

Technical report LERSSE-TR-2006-03[†]

December 17, 2006

### Abstract

Security administrators prevent security breaches against their infrastructure by using their tools to implement the security policy. This paper deals with security administration errors that were collected from the RISKS-forum and were analyzed using grounded theory. The application of open coding, one of the components of grounded theory, led to a classification of errors based on security tasks. Security errors were also divided according to whether the error was due to Human limitations, Organizational limitations, Technological limitations (HOT) or a combination of these limitations. Moreover, security administration errors were categorized according to different functionality. Our findings have pointed out that security administrators commit a variety of "configuration" errors as well as errors that fall under the category of "patching and upgrading." We also encountered one error under the category of "password maintenance." Our results showed that human limitations played a crucial role in the errors that we logged in this study. Thus, we have recommended that more study needs to be conducted into the human factors of security administration.

[*] kartikm@ece.ubc.ca

[†] This and other LERSSE publications can be found at lersse-dl.ece.ubc.ca

# Contents

# 1 Introduction

Institutions and corporations depend upon security administrators to implement their security policy and prevent security breaches. Despite its importance, security administration has largely been ignored by the research community until recently. An ethnographic study by Kandogan and Haber had sought to reveal the work habits of security administrators [4]. In this paper, we shed further light on this issue by studying the errors committed by security administrators.

The problems that we have investigated in this study are two-fold. Firstly, we wanted to understand the types of error made by security administrators. Secondly, we wanted to classify errors according to the Human, Organizational and Technological (HOT) limitations[1]. The problems that we explored are important because they provide researchers with insights on the security adminstration issues at hand. Moreover, this study lays the foundation towards mitigating future security administration errors.

The approach we took to address our problem was to gather security administration errors from the RISKS-forum. The RISKS-forum[2] offered a unique opportunity to collect anecdotal data as well as news reports of risks. By reading through the RISKS-forum postings we selected and developed a list of security administration errors. We then analyzed our data by using grounded theory. Through grounded theory we categorized our error into three groupings: "configuration", "password maintenance" and "patching and upgrading." Most of the errors belonged to the "configuration" category. When we classified errors according to security functionality we encountered errors due to access control, authentication and availability. Finally, our data showed us that security administration errors involve a human limitations which could be attributed to cognitive constraints or carelessness. Furthermore, future research should outline the human factors for each of the tasks performed. Then, these should be integrated into the development of security tools.

The rest of the paper is organized as follows: Section 2 introduces the methods that underlie our research. Section 2.1 describes the sources of data that was used for this study. Section 2.2 elaborates on the selection criteria that was used to classify errors. Section 2.3 discusses the field values in the spreadsheet that we developed. Section 2.4 addresses how we analyzed our data. Section 3 describes the results of our data. Section 4 points out the implication of our results. Section 5 discusses the related work in security administration. Section 6 summarizes and puts forth recommendations. Appendix A cites the risk references that were used in this paper. Appendix B cites the references we used in our spreadsheet.

# 2 Methods

## 2.1 Sources of data

For this study we catalogued errors from the RISKS-forum, which included both anecdotal and published accounts of security administration errors. The RISKS-forum is moderated by Neumann and is a public digest for reporting risks. It was established in 1986 and continues to this date. Individuals post submissions that describe newspaper

[1] HOT classifications were developed Beznosov et. al and are explained on page 4

[2] http://catless.ncl.ac.uk/risks

and magazine accounts of risks. Personal experiences with risks are also posted. Risk is defined by the Webster's dictionary as the chance of hazard [6].

We tried to address validity issues regarding the sources for our data. We were unable to assure the accuracy of anecdotal data. For other data, if there was no hyperlink available or if the source was not accessible, then we conducted a web-search. However, for the early issues of the RISKS-forum, the information was simply not available on the internet.

Often the data collected for qualitative research originates from observations, semi-structured interviews and focus groups. However our only data source stemmed from RISKS-forum postings. Thus, we were unable to triangulate [5] and solidify the validity of our results. Triangulation is a process where different data sources such as interviews and observations are considered in addition to the posting in order to better understand the errors made by security administrator. These constraints made it impossible to answer the question as to exactly why an error took place. We often guessed as to which HOT classification or combination contributed to the error. However, we had the opportunity to address the types of error that occurred.

## 2.2 Selection criteria

Our selection criteria necessitated a definition of key terms:

**Definition 1 (Security Administration)** *Security administration is a role that involves the configuration, maintenance, testing and installation of the technology used to enforce an organization's security policy. Recovery from malicious actions and response against attacks are also key tasks that are performed. This definition is inspired from Barret et al.'s definition of security administration [1].*

**Definition 2 (Security Administrator)** *Security administrator is an individual who performs the security administration role on behalf of the organization.*

**Definition 3 (End User)** *End user is an individual who performs security administration role on his or her own behalf.*

**Definition 4 (Security Administration Error)** *Security administration error is the inability of an individual to properly use the technology to enforce the security policy when performing a security administrative task.*

Our selection criteria involved a five step decision making process that analyzed whether a security administration error (see Definition 4) took place. In the first step, we examined whether the incident under review involved the performance of a security administrative task (see Definition 1). If so, we investigated whether the error could have been caused by a security administrator (see Definition 2). In the third step, we explored whether a security administration error might have occurred (see Definition 4). We did this by examining evidence that suggested that a security administration error took place. In the fourth step, we actively searched for evidence that pointed to alternative explanations for the outcome described by the incident. In the final step, if the explanation in favor of a security administration error outweighed any of the alternative explanations then we catalogued the incident under the title "Record of security administration errors". However, if the explanation in favor of security administration error weighed equally with other alternative arguments then we added the incident to another list titled "Record of inconclusive security administration errors". On the other hand, if the alternative explanations outweighed the argument in favor of the security administration error then the incident was not catalogued.

The security policy, which separates secure states from insecure states, played a key role in deciding if a security administration error took place. The security policy is created by the management; the security administrator then implements the policy on behalf of the organization. The security policy can be explicit or implicit. However, in both the implicit and explicit cases the security policy serves to protect the confidentiality, integrity and availability of sensitive data and resources [2]. When cataloguing security administration errors we tried to first determine if there was any evidence that suggested a breach in the policy. If there was no evidence that suggested that a breach occurred then we did not catalogue the incident as an error. If we were able to deduce that a breach occurred we logged the error.

When selecting alternative arguments that can be used to explain the outcome of a particular incident we considered the tradeoffs that were made by the administrator in enforcing the policy. A common tradeoff made by an administrator was to sacrifice security for cost. For example, a security administrator might not have invested in an encryption software due to budget restraints even though the encryption software would have provided an added layer of protection (RisksRef. 1). When tradeoffs such as these were made we did not log them as errors in our "Record of security administration errors" because they did not breach the existing security policy.

Other sources that can contribute to the outcome described by the incident include software bugs and end user errors. An error in a software component used by the administrator could contribute to the malfunction of a system. In one of the cases (RisksRef. 2) we considered whether the security policy was breached because the webpage was not securely programmed. The fault lay in the improper design or the implementation of the webpage rather than in the configuration of the webpage by the security administrator. Due to this reason, we did not catalogue the incident as an error. In general, we ruled out any software design and implementation errors.

We also tried to eliminate any errors made by end users (see Definition 3). In one example, an end user lost all of the encrypted data because he deleted the private key (RiskRef. 3). In this case, the individual was performing the encryption on behalf of himself. Thus, we did not log the incident as a security administration error. In essence, each case incident was analyzed to see if the error was caused by a security administrator. If the task was done on behalf of an organization then it was considered as a security administration error, if not we declared it as an end user error.

We encountered several problematic incidents that we could not conclusively resolve with the information that was provided in the RISKS-forum. For such cases, we logged these incidents into the "Record of inconclusive security administration errors". Take, for example, a case (RiskRef. 4) where an FTP server had research information in a directory labeled "research:" as well as directory listing of .login, .cshrc and a .rhosts file. The permissions were set so that anyone could have downloaded any information that was available on the web server. Although at first sight this looks like a security administration mistake it was not clear if the security policy was violated. There might not have been a policy that covers how the FTP server should have been configured. There were also cases where we were unable to pinpoint whether a security administrator or end user made an error. In one case (RiskRef. 5), an e-mail distribution list was left unsecured and intruders were able to get the emails of people on the list. From reading the information in the security incident it was not clear if an end user made the error. Thus, we entered these two incidents in the "Record of inconclusive security administration errors."

Although software testing was included in our definition of the security administrative role, it was not easy to attribute errors to testing. Take, for example, a case where a software upgrade caused a security breach in the production environment whereby customers who logged into the website were able to read other peoples' email (RiskRef. 6). It was hard to declare this a security administrative error due to poor testing. This was because there might not have been a policy in place to test the upgrade. Even if a policy had existed, the source of the error could be due to the lack of a staging environment that mirrored the production environment. Furthermore, as Djikstra points out it is common knowledge that "Program testing can be used to show the presence of bugs, but never to show their absence" [2].

## 2.3 Field values

This section provides details of different field values that we recorded about each selected incident.

**Incident short title:** we provided a two to five word description of the error. Example: Security patch was not installed (RiskRef. 7).

**Description of the Error:** a short synopsis of the error was provided. Example: Individuals in each division were responsible for installing security patches, a software program that counters hackers. With no centralized security group monitoring the work, the legislative auditor found that patches weren't installed on many computers (RiskRef. 7).

**Security functionality:** the error was classified according to one of the security functionalities. Example: Access control (RiskRef. 7).

**HOT classification:**

- **Human:** Human errors are those that resulted from cognitive barriers and other limitations of humans. In this paper, we also considered carelessness and lack of motivation as sources for human errors. Some examples include: attention deficit, perception errors, lack of memory, etc.
- **Organizational:** Organizational errors are those that result from organizational limitations. Examples of this include lack of manpower, budget constraints, poor management, etc.
- **Technological:** Technological errors are those that result from technological limitations. Technological limitations indicate the problems posed by current technology. Examples: Poor usability, poor visualization, etc.

**Rationale:** the rationale for the HOT classification was provided. An error was classified as technological if the error was rooted in the technology and not in the human being using the technology. More specifically, a technological error would have occurred even if a perfect human being, with no human limitations, was performing the task. On the other hand, if the error was committed due to human factors then the error was labeled human. Finally, an error was classified as organizational if a perfect administrator committed a mistake due to organizational demands. If an incident did not meet a criteria exactly we made an educated judgement as to which category the error belonged to. Example of a combination of human and organizational error: Security administrators did not install the patch at the correct time. It seems that there were also cost cuts which limited what administrators could do. Furthermore, a couple of administrators left the job. Security consultants were not hired (RiskRef. 7).

**How the error was found:** we outlined the procedure detailing how the error was first discovered. Also, if the information was unavailable then this field was marked "unknown." Example: The error was found through two legislative audits (RiskRef. 7).

**Action taken in response to error:** we described the steps taken by both the administrator and the organization after the error was discovered. If this information was unavailable then it was marked "unknown." Example: Students were punished (RiskRef. 8).

**Description of effort to recover from errors:** we described the effort needed to recover from the error. If the information was unavailable it was marked "unknown." All of our incidents were marked as "unknown."

**Impact:** we provided a statement of the repercussions of the error on the organization. Example: The credibility of the weather service was at stake because of the wrong announcement of the actual weather. Example: Data may have been lost (RiskRef. 7).

**Time taken to recover from error:** the length of time taken to recover from the error was provided. If the time was unavailable then it was noted "unavailable." Example: 2 days (RiskRef. 9).

**RISKS-forum posting source:** we reported the reference to the RISKS-forum posting of the article. Example: Peter G. Neumann, neumann@csl.sri.com, "Security hole at WorldCom left internal computer networks at risk", Volume 21, Issue 81, December 6, 2001 (RiskRef. 7).

**RISKS-forum hyperlink:** the hyperlink of the RISKS-forum posting was provided.

**Other sources:** we reported the reference to the source of the article and the corresponding RISKS-forum post. Example: "WorldCom network security hole fixed with help of hacker", Todd R. Weiss, Computerworld, December 6, 2001 (RiskRef. 7).

**Other sources hyperlink:** the hyperlink of other sources were provided.

**CIA:** we reported whether the error was a breach of confidentiality, integrity or availability. Example: Confidentiality and integrity (RiskRef. 7).

**Date error was researched:** we reported the date we recorded the error in our database. Example: 8-Oct-05 (RiskRef. 7).

**Start date of error:** we recorded the date when the error first occurred. If the start date was not provided in the incident then we recorded the date as January 1, 1900. Example: 1-Oct-00 (RiskRef. 7).

**End date of error:** we recorded the date the error was fixed, in the case that the error was not fixed we provided the date the error was researched. Example: 19-Apr-05 (RiskRef. 7).

**Date error was reported:** we recorded the date when the error was first published. If the RISKS-forum alluded to another article as it's source then the date of that article was recorded. If not, the date of the RISKS-forum was recorded. Example: 2-Oct-05 (RiskRef. 7).

**Alternative explanations:** this field was used in the "Record of inconclusive security administration errors" to provide alternative arguments that could support the outcome described by the incident. Example: The error might have been caused by an end user (RiskRef. 10).

## 2.4 Data analysis

We used grounded theory to analyze our data. Grounded theory was discovered by Glaser and Strauss [3] whereby the actual theory emerges from the data. Grounded theory is different from hypothesis testing where the researcher's goal in hypothesis testing is to use the data to verify or reject the hypothesis. We used grounded theory because we wanted to formulate our understanding of the security administration errors directly from the security incident reports. Open coding [3] was the only method we used in grounded theory to analyze the data that we collected. Open coding involved developing categories according to common themes that emerged from the data. After reading the security incident, we used open coding to mark and label errors according to the different tasks that led to the error performed by the security administrator. We also labeled errors with respect to their security functionality and HOT classification.

# 3 Results

We counted the number of errors that belonged to each security functionality. We found that eight belonged to access control (B 1, 4, 15, 16, 18, 23, 24, 19), five belonged to authentication (B 8, 10, 25, 33, 2), and one belonged to availability (B 26). Access control errors indicated that security administrators were unable to uphold the security policy whereby access was only granted to selective individuals. Errors that were classified in this paper as belonging to access control dealt with cases where information that was thought to be secure was either leaked or potentially could have been leaked to a third party. Authentication errors that were observed in this paper were due to login procedures that were either poor, nonexistent, or were thwarted by adversaries. The error attributed to availability was the result of a security administration error which could have resulted in the loss of data.

When we classified errors with respect to HOT terminology we found that twelve errors occurred due to human error (B 2, 4, 8, 10, 15, 16, 18, 19, 23, 24, 25, 33), while one error occurred because of human limitations and organizational demands (B 1), and one due to human limitations and technological difficulties (B 26). The human errors could be attributable to either a lack of motivation or due to cognitive limitations. The error due to human limitations and organizational demands was attributable to a situation where the security administrators in the organization were overworked and did not have the time to perform all their tasks. The error due to human limitations and technological difficulty could be credited to the poor usability of the security tools.

Through open coding we partitioned our security incidents into three main categories. We then counted the number of occurrences in each category. We found out that there were three errors that fell under the area of "patching and upgrading" (B 1, 10, 25). Only one error was attributable to the area of "password maintenance" (B 2). Ten errors fell under the umbrella of a category named "configuration." We further divided the "configuration" category into errors that occurred due to "configuring certificates" (B 19, 23) and "configuring systems." We counted two errors that were attributable to "configuring certificates" while eight were attributable to "configuring systems." The "configuring systems" (B 24, 26, 33, 18, 16, 4, 15, 8) category consisted of case incidents where the errors arose when security administrators configured: a router, virus scanner, login information, operating system, permissions of a log file, firewall and a database. We encountered two errors that were due to faulty configuration of firewalls.

# 4 Discussion

This research is a starting point for future work in analyzing the errors made by security administrators. Through categorizing errors according to different functionality we showed the variety of we encountered. A majority of these errors belonged to access control, however this does not mean that most errors made by security administrators are due to access control. Instead, it simply reflected the large number of access control errors that we encountered in our data. Through studying our security incident reports future investigators could understand the variety of access control errors, authentication errors and availability errors.

All of the errors that we collected and classified according to HOT terminology had a human component. This meant that errors arose out of a result of two factors: carelessness or cognitive limitations. Unfortunately, we were unable to determine as to which one of these two they were. By resolving which of the two factors the error belonged to we could answer whether current security tools meet the needs of security administrators. If the error occurred due to cognitive or other human limitations then we could find out which tools pose problems, what problems they pose and recommend better ways to address these problems. On the other hand, if the error occurred due to carelessness we cannot make any of these inferences. Our research would have benefited from observations and interviews which would have allowed us to better understand the technology used by security administrators as well as to resolve which of the two factors led to the error.

Our categorization of different security tasks allows future researchers a chance to find out the causes of these errors. We encountered several different "configuration" errors. In all of the cases the administrator did not achieve his intention while using the tools that were available to him or her. Unfortunately, our research cannot answer why the error occurred. Instead, our research has shown that problems exist in security administration. It has shown that security administrators make errors when using their tools to configure systems in order to enforce the security policy. Our study also showed that security administrators were unable to track and install new patches that were issued by the vendor. Moreover, our "password maintenance" error has pointed out that security administrators sometimes did not follow common security guidelines such as not writing down one's password and not generating a password that can be easily guessed.

# 5 Related work

Kandogan and Haber [4] have utilized ethnographic research to study the lifestyle of security administrators in a university environment. In particular they profiled a security manager and an engineer through interviews and observations. They used grounded theory to analyze their observations. Our research adds to their body of work and offers further insights into security administration. The focus of our work is on understanding the type of errors made by security administrators as well labeling these errors according to the HOT terminology.

Kandogan and Haber have pointed out the various tasks performed by a security administrator; these range from tackling viruses, using intrusion detection software, real-time network monitoring, performing security scans and safeguarding their organization from attacks by hackers. Kandogan and Haber pointed out that the tools used by security administrators have several problems. In particular, they have asserted that the security tools are not well integrated with one another. Moreover, they have argued that security administration is a collaborative environment; however the tools used are not fully co-operational. Perhaps, these reasons could potentially explain the causes of some of the errors we encountered in our research. Kandogan and Haber have gone on to conclude that better security tools need to be created. Our research has indicated that before such tools are developed there needs to further study of the human factor in security administration.

# 6 Summary and recommendations

We have tried to get an understanding of the different errors made by security administrators. We did this by analyzing reports of security incident from the RISKS-forum. We used qualitative research to classify errors according to different criteria. We first classified the errors according to different security functionalities, then we classified errors according to the HOT terminology. Finally, we classified errors based on the nature of the task performed by a security administrator.

These errors could be reduced through the application of two key principles for designing secure software, namely security in depth and fail-safe defaults [2]. Through security in depth, a single careless mistake by the administrator will not result in a breach. Furthermore, through fail-safe defaults access will not be guaranteed unless explicitly specified. Thus, if the system was initially in a safe state then it will continue to be in that state.

# References

[1] R. Barrett, E. Haber, E. Kandogan, P. Maglio, M. Prabaker, and L Takayama. Field studies of computer system administrators: Analysis of system management tools and practices. In *Proceedings of the Conference on Computer Supported Collaborative Work*, 2004.

[2] Matt Bishop. *Introduction to Computer Security.* Addison Wesley, Boston, MA, 2005.

[3] Barney Glaser and Anselm L. Strauss. *The Discovery of Grounded Theory, Strategies for Qualitative Research.* Aldine Publishing Company, Chicago, Illinois, 1967.

[4] Eser Kandogan and Eben M. Haber. Security administration tools and practices. In Lorrie Faith Cranor and Simson Garfinkel, editors, *Security and Usability: Designing Secure Systems that People Can Use*, chapter 18, pages 357–378. O'Reilly Media, Inc., Sebastapol, 2005.

[5] J. Maxwell. *Qualitative Research Design: An iterative approach.* Sage Publications, California, U.S.A, 2005.

[6] Merriam-Webster. Merriam-webster's collegiate dictionary, 1994.

# Appendix A Risk forum references

Below are RISK-forum references used in this paper:

RiskRef. 1: Neumann, P.G., neumann@csl.sri.com, "Time Warner backup tapes lost with 600,000 records", May 2, 2005, Vol. 23, Issue 86.

RiskRef. 2: Drew Dean, ddean@csl.sri.com, "Victoria's Secret Reaches a Data Privacy Settlement", Volume 21, Issue 97, October 21, 2003.

RiskRef. 3: Herman D. Knoble, hdk@psu.edu, "Risk of backing up PGP Key Ring files", Volume 20, Issue 30, April 7, 1999.

RiskRef. 4: John P. Wilson, jowilson@mtu.edu, "Major corporation's misconfigured FTP server", Volume 19, Issue 22, June 11, 1997.

RiskRef. 5: Declan McCullagh, declan@well.com, "European Commission "Net-security" site invaded by hackers", Volume 21, Issue 48, June 13, 2001.

RiskRef. 6: Pete Morgan-Lucas, pjml@nsgmail.nerc-swindon.ac.uk, "Barclays Internet-banking security-glitch following software upgrade", Volume 21, Issue 1, August 1, 2000.

RiskRef. 7: Steven Hauser, hause011@tc.umn.edu, "Audit shuts down Minnesota Car License Web", Vol: 18, Issue: 85, April 19, 2005.

RiskRef. 8: Thom Kuhn, tkuhn@mail.acponline.org, "Students Face punishment for computer tampering", Vol: 24, Issue: 2, August 10, 2005.

RiskRef. 9: Peter G. Neumann, neumann@csl.sri.com, "Security hole at WorldCom left internal computer networks at risk", Volume 21, Issue 81, December 6, 2001.

RiskRef. 10: Paul Kafasis, paul@pbones.com, "E-mail hoax at University of Maryland", Volume 22, Issue 72, May 4, 2003.

# Appendix B Record of errors

Below are the serial number and the references for each of the errors that were recorded in the Record of errors spreadsheet:

| Serial Number | Reference |
|---|---|
| 1 | Steven Hauser, hause011@tc.umn.edu, “Audit shuts down Minnesota Car License Web”, Vol: 18, Issue: 85, April 19, 2005. |
| 2 | Thom Kuhn,tkuhn@mail.acponline.org, “Students Face punishment for computer tampering”, Vol: 24, Issue: 2, August 10, 2005. |
| 4 | David Kennedy, david.kennedy@acm.org, “Errant weather alert”, Vol: 23, Issue: 13, January 13, 2004. |
| 8 | Peter G. Neumann, neumann@csl.sri.com, “Defense Information System Agency leaves shopping list online”, Volume 22, Issue 28,October 2, 2002. |
| 10 | Monty Solomon, monty@roscom.com , “Crackers steal 52,000 university passwords”, Volume 22, Issue 39, November 20, 2002. |
| 15 | Fuzzy Gorilla, fuzzygorilla@euroseek.com, “Slammer worm hits system within Davis-Besse nuclear power plan”, Volume 22, Issue 88, August 22, 2003. |
| 16 | Esteban Gutierrez-Miguel, esteban@ce.net.mx, “The Globe and Mail* Web site exposing search-engine log file”, Volume 21, Issue 2, August 17, 2000. |
| 18 | Declan McCullagh, declan@well.com, “European Commission ”Net-security” site invaded by hackers”, Volume 21, Issue 48, June 13, 2001. |
| 23 | Jeremy Epstein, jepstein@webmethods.com, “Even professional organizations forget about certificate expiration”, Volume 21, Issue 74, November 5, 2001. |
| 24 | Peter G. Neumann, neumann@csl.sri.com, “Security hole at WorldCom left internal computer networks at risk”, Volume 21, Issue 81, December 6, 2001. |
| 25 | Monty Solomon, monty@roscom.com, “Stanford e-mail system passwords stolen”, Volume 20, Issue 5, November 4, 1998. |
| 26 | Diomidis Spinellis, dspin@aegean.gr, “Virus cleaner corrupts email database”, Volume 20, Issue 40, May 18, 1999. |
| 19 | Debora Weber-Wulff, weberwu@fhtw-berlin.de, “Berlin Bank shows sensitive information” , Vol 21, Issue 50, July 9, 2001. |
| 33 | Jerry Leichter, leichter-jerry@cs.yale.edu, “VMS and login failure logins”, Volume 6, Issue 16, January 27, 1988. |