When Convenience Trumps Security: Defining Objectives for Security and Usability of Systems

Gurpreet Dhillon, Tiago Oliveira, Santa Susarapu, Mário Caldeira

When Convenience Trumps Security. Defining Objectives for Security and Usability of Systems

Gurpreet Dhillon[1], Tiago Oliveira[2], Santa Susarapu[1], and Mário Caldeira[3]

[1]School of Business, Virginia Commonwealth University, Richmond, VA, USA
{gdhillon, susarapusr}@vcu.edu
[2]ISEGI, Universidade Nova de Lisboa, Portugal
toliveira@isegi.unl.pt
[3]ISEG, Technical University of Lisbon, Portugal
caldeira@iseg.utl.pt

Abstract. Security and usability of systems continues to be an important topic for managers and academics alike. In this paper we propose two instruments for assessing security and usability of systems. These instruments were developed in two phases. In Phase 1, using the value-focused thinking approach and interviews with 35 experts, we identified 16 clusters of *means* and 8 clusters of *fundamental* objectives. In phase 2 drawing on a sample of 201 users we administered a survey to purify, ensure reliability, and unidimensionality of the two instruments. This resulted in 15 means objectives, organized into four categories (*minimize system interruptions and licensing restrictions*, *maximize information retrieval*, *maximize system aesthetics*, and *maximize data quality*) and 12 fundamental objectives grouped into four categories (*maximize standardization and integration*, *maximize ease of use*, *maximize system capability*, and *enhance system related communication*). Collectively the objectives offer a useful basis for assessing the extent to which security and usability has been achieved in systems.

Keywords: security values, usability values, value focused-thinking, qualitative methods, instrument development, quantitative methods.

1 Introduction

Consider a situation where Alice has to set up her friend's new computer. Alice sets up a limited user account, changes the file permissions for the entire user class, thus allowing *write* and *modify* access for all non system partitions. Alice goes a step further and installs a freeware *SuRun* so that in case her friend called her again, she could run certain programs without necessarily asking for administrative rights. Alice also installed *Returnil*, a software suite that allows automatic recovery in case of problems. All in all, Alice considered this to be a rather secure and a usable arrangement. Alice's friend however encountered significant problems with the set up. Following a system update, SIW (System Information for Windows) kept popping up warning windows and later hung up. With little computing knowledge Alice's friend tried uninstalling the virtual machine. With some luck she was able to delete

the folder, which stopped the conflict messages from popping up. However in the process she probably left her computer open for several vulnerabilities.

So, where does one draw a line between security and usability? Alice thought that she was probably doing a pretty good job, but by imposing her own values in configuring a system, she ended up creating several vulnerabilities without even realizing that they existed. Within organizations it is common to see such problem occur and the solution perhaps resides in addressing user expectation and inferring authorizations based on designations [1]. While the literature has made several calls for balancing security and usability, these have not been adequately heeded to (e.g. see [2, 3]). In majority of the cases where security and usability has been considered, it's been an afterthought at best with developers design systems and later realizing that security and usability considerations had not been adequately addressed. Such development duality typically results in a haphazard system development (see [4, 5]). A definition of a common set of security and usability objectives would to a large extent elevate the development duality problem by presenting objectives that must be achieved. Such objectives would also help in providing a strategic direction for secure and usable systems development.

In this paper, following Keeney [6], Dhillon and Torkzadeh [7], and Torkzadeh and Dhillon [8], we use value-focused thinking to define security and usability objectives. The study was undertaken in two phases. Phase 1 involved a qualitative definition of value-based objectives. Phase 2 helped in developing a parsimonious set of security and usability objectives. The two phases and a final set of objectives are discussed in the rest of the paper.

2 Value Focused Security and Usability Objectives

As stated earlier, methodologically this research is based on Keeney's [6] 'value focused thinking' approach. Keeney suggests that most decision-making methods are based on alternative thinking practices where choices are made only from a limited list of available alternatives. The alternative based approach is constrained by the limits imposed by decision-makers in the process of identifying constraints and subsequently alternatives. As a consequence, individuals tend to forget what they really want to achieve. Since achieving an objective is the primary reason for being involved in any decision situation, Keeney contends that one should remain focused on the bottom-line objectives, which makes decisions meaningful and of value, instead of making choices among current alternatives. Value focused thinking is proposed as a method by Keeney, to address the most fundamental question - what do we want to do and why. Research conducted by Keeney (e.g. see [6, 9]), has attempted to expose underlying values in a wide array of decision contexts. The inherent argument is that the value thinking process can help researchers and managers alike to be proactive and hence create more alternatives instead of being limited by available choices.

Value focused thinking consists of two main steps in eliciting and framing values: (1) conduct interviews and construct a list of what they want in the decision context, (2) convert these statements into a common format of objectives (an object and a

preference). In terms of modeling the means and fundamental objectives, a network hierarchy can also be put together. In the context of our research, this two-step method is applied in order to assess values attached by users, to IS Security and Usability. Thirty-five end-users of IS/IT services were contacted from employees of five large businesses in the US. The businesses represented the following industries – IT consulting, Hotel and Casino, Banking, Education and Training. The interviews formed the basis for eliciting the values.

Construct a list of what users want. The best way to find out what users value most is to ask them. Also, it is better to ask as many users as possible because different users may have different values and they may express them differently. However a difficulty lies in the latency of these values. In many cases users' values are hidden under the surface. Keeney recommends several stimulation techniques to surface these latent values. We chose a combination of two techniques to identify the latent values. The first method used was a wish list. Each interviewee was asked to express what their needs were in terms of security and usability of systems they used within their organizations. The second method, which augments the simple wish list method, was the probing technique. In order to expand the wish list and whenever subjects are having problem articulating what they want, interviewer posed several probing questions prepared beforehand. The list of probing questions included: “If you did not have any constraints, what would your objectives be?” "What needs to be changed from the status quo?" "How do you evaluate security and usability of systems?" "What do you expect in terms of security and usability?" "How do they tell if security and usability of systems is good or bad?" Besides asking the interviewees to generate a wish list, we also asked them to generate a list of problems and shortcomings in security and usability of systems they used. The basic idea behind asking problems and shortcomings was to generate objectives by articulating their concerns. The thirty-five interviews generated three hundred and thirty seven wishes/problems/concerns.

Convert statements into objectives. These statements are converted to objectives, using a verb (direction of change) plus an object (target of change) format. Some statements on the list are compound sentences, which produce more than one objective, and some statements were being repeated by several users. For example, one user wishes "to be educated in moving between different applications and wants help when he gets lost." Two objectives can actually be derived from this wish: (1) ease of navigation through the application and (2) enhance system training quality. To eliminate these ambiguities and redundancies, two researchers reviewed each item on the list independently. This review and refinement produced one hundred and thirty objectives in a common form of a verb plus an object.

In ensuring security and usability, users wanted to achieve these one hundred and thirty objectives. However, these objectives are not adequately articulating values yet, and also include duplication. The objectives were then categorized in order to surface the meanings, and the values attached to cluster the objectives. The categorization resulted in twenty-four clusters of objectives.

As a next step of framing values out of objectives, twenty-four objectives were classified into two categories: means objectives and fundamental objectives. The criterion of classification is whether an objective is an intermediate one, i.e. is it a means to achieve another objective or is it a final and a fundamental one in terms of

security and usability. As a result, eight fundamental objectives were identified. The means objectives, a total of sixteen, are presented in Table 1 and fundamental objectives in Table 2. The tables do not show all the individual objectives. These are available from the authors on request.

Table 1. Means objectives

Means Objectives (16 clusters, 91 items)	
Clarify & improve system documentation e.g.: Ensure easy access to system documentation	**Maximize system access** e.g.: Define role-based external access
Improve system search capability e.g.: Ensure semantic based search features	**Maximize system efficiency** e.g.: Ensure process fairness
Maximize data quality e.g.: Enhance data integrity	**Maximize system esthetics** e.g.: Enhance visualization of system security
Maximize database and system access e.g.: Ensure web access to the system	**Maximize system integrity** e.g.: Maximize system adaptability
Maximize disaster recovery e.g.: Ensure data availability	**Maximize system maintainability** e.g.: Ensure hardware robustness
Maximize productivity e.g.: Ensure automated password retrieval	**Maximize system reliability** e.g.: Maximize process execution accuracy
Maximize security & privacy e.g.: Decrease restrictiveness of system	**Maximize task efficiency** e.g.: Maximize automation of manual tasks
Maximize self-efficacy in training e.g.: Enhance system training quality	**Minimize system interruptions** e.g.: Minimize system down-time

Table 2. Fundamental objectives

Fundamental Objective (8 clusters, 59 items)	
Enhance system related communications e.g.: Ensure exception reports go to management	**Maximize system administration functionality** e.g.: Enhance connectivity at affordable price
Improve data organization e.g.: Ensure data archival functionality	**Maximize system capability** e.g.: Enhance application features
Maximize ease of use e.g.: Ensure ease of navigation through application	**Maximize system integration** e.g.: Ensure functionality is designed into system
Maximize standardization of system features e.g.: Enhance customizable interfaces	**Maximize user requirements elicitation** e.g.: Ensure system functionality meets requirements

3 Quantitatively Derived Parsimonious Set of Security and Usability Objectives

3.1 Method

In Phase 1, 150 items that influence information systems (IS) usability were developed. These items were based on the total set of 130 objectives identified in phase 1. The additional 20 items were added to ensure that all objectives were well represented in the survey instrument. These items were grouped into two categories of means and fundamental objectives. The means objectives contain 91 questions (items) grouped in 16 clusters (constructs). The fundamental objectives present 59 items grouped in 8 constructs. The large number of items found in both objective sets may have led to redundancy, but it helped content validity, since we were drawing on a large universe of possible items [10].

Based on items found in Phase 1, we developed a questionnaire. A five-point Likert-type scale was used, with a range from one (strongly disagree) to five (strongly agree). The respondents were asked to express agreement with 150 questions pertaining to the following context statement - *"In order to respond to the questions below, think of any system that you may be using or are familiar with. What would your ideal state be in terms of achieving your objectives?"* The survey was administered to graduate and undergraduate students in a European University. We obtained a sample of 201 (30.3% male, 69.7% female) respondents. The respondent rate was 66.3%. The respondents were mature students with work experience in a variety of professions, such as, banking, sales, healthcare, information systems, engineering, education among others areas. The age of the participants ranged from 18 and 50. All participants had experience with security and usability of IS, thus being qualified to answer this survey.

The analysis of the data was undertaken with several goals: purification, reliability, and unidimensionality. The following three steps were used in the elimination process:

1. We eliminated the items if their corrected item-total correlation (the correlation of each item with the sum of the other items in its category) was less than 0.5, because, according to Churchill [11], all items that belong to the same domain of the concept (construct) should be highly inter-correlated.
2. We eliminated the items if the reliability of the remaining items was at least 0.9. Cronbach's α was computed to see if additional items could be eliminated without substantially lowering reliability.
3. A factor analysis was undertaken with the remaining items for each group to eliminate items that were not factorially pure [12]. This means that we eliminated items that had a loading greater than 0.3 on more than one factor.

The purification of the items was done before the factor analysis, to produce less dimensions and not to confound the interpretation of the factor analysis [11]. This methodology provides brevity and simplicity of the factor structure.

3.2 Data

Means Objectives. We performed the item procedure, described before, to purify the means objectives category. First, corrected item-total correlation of less than 0.5 suggests the elimination of 29 items. Second, the reliability analysis does not eliminate any item. Finally, the factor analysis suggests an elimination of 47 more items.

After the elimination process, 15 items of the means objectives category were obtained. In Table 3, we present the results of the factor analysis using varimax rotation for the retained items. Bartlett's test of sphericity was 1278.4 ($p < 0.001$). This means that the data contains enough common variance to perform a factor analysis. Kaiser-Meyer-Olkin (KMO) measures the adequacy of the sample; KMO is 0.88 (KMO $\geq$ 0.80 is good [13]), which reveals that the matrix of correlation is adequate for the factor analysis. The results of the factor analysis revealed four factors with eigenvalues greater than one. These factors explain 67.5% of the variance contained in the data.

Table 3. Factor analysis of means objectives in Phase 2 (n=201)

	F1	F2	F3	F4	Corrected Item-Total Correlation
Minimize system interruptions and licensing restrictions					
Minimize unnecessary system lock outs & time outs	0.73				0.69
Minimize system interruptions	0.73				0.64
Minimize application licensing restrictions	0.71				0.64
Minimize the total cost of ownership	0.63				0.63
Minimize system down-time	0.55				0.55
Maximize information retrieval					
Maximize efficiency of system tasks		0.74			0.74
Maximize task efficiency		0.73			0.71
Maximize system efficiency		0.71			0.72
Maximize database and system access		0.62			0.65
Maximize system esthetics					
Ensure color combinations are visually appealing			0.81		0.66
Ensure good application display			0.68		0.66
Maximize system esthetics			0.59		0.60
Maximize data quality					
Enhance data integrity				0.74	0.61
Increase timely application data access				0.68	0.59
Increase ease in editing and updating of application data accurately				0.56	0.53
Eigenvalue	5.77	1.70	1.38	1.26	-
% Variance	38.5%	11.3%	9.2%	8.4%	-

Note: loadings greater than 0.3 are reported; the items are grouped by highest factor loading and presented by descending order.

The four factors found were easily interpreted, they are: *minimize system interruptions and licensing restrictions* (five items), *maximize information retrieval (*four items), *maximize system esthetics* (three items), and *maximize data quality* (three items). The range of loadings is respectively: 0.55-0.73, 0.62-0.74, 0.59-0.81, and 0.56-0.74. All the factors have a loading greater than 0.5. This indicates that our analysis employs a well-explained factor structure.

The range of corrected item-total correlation varies between 0.55 to 0.69 for *minimize system interruptions and licensing restrictions*, 0.65 to 0.74 for *maximize information retrieval*, 0.60 to 0.66 for *maximize system esthetics*, and 0.53 to 0.61 for *maximize data quality*.

The reliability for each construct was: 0.84 for *minimize system interruptions and licensing restrictions*, 0.86 for *maximize information retrieval*, 0.80 for *maximize system esthetics*, and 0.75 for *maximize data quality.* The overall reliability for the 15 item scale was 0.88. This reveals that reliability exceeds the suggested cutoff value of 0.70 [14].

Fundamental Objectives. To purify the fundamental objectives category we used the same item purification procedure. The first criteria was to eliminate items below 0.5, it allowed us to eliminate 17 of the 59 items obtained in Phase 1. The second criteria, reliability analysis, did not eliminate any items. Finally, the factor analysis suggested the elimination of 30 more items.

Subsequent to the elimination process, the fundamental objectives scale included 12 items. First, Bartlett's test of sphericity was 1292.3 ($p < 0.001$). These factors explain 76.5% of the variance contained in the data. The KMO is 0.82 ($KMO \geq 0.80$ is good [13]), which reveals that the matrix of correlation is adequate for factor analysis. This reveals that the data contains enough common variance to perform the factor analysis. Four factors with eigenvalues greater than one is obtained (Table 4), and the interpretation of each factor was not difficult, i.e.: *maximize standardization, integration and user requirements* (4 items), *maximize ease of use* (3 items), maximize system capability (3 items), and *enhance system related communications* (2 items). The ranges for factor loading were, respectively, 0.61-0.86, 0.56-0.85, 0.62-0.75, and 0.87-0.87. All the factors have a loading greater than 0.5. This indicates that our analysis employs a well-explained factor structure.

The range of corrected item-total correlation for each item varies between: 0.61 to 0.78 for *maximize standardization, integration and user requirements*, 0.59 to 0.74 for *maximize ease of use*, 0.61 to 0.72 for maximize system capability, and 0.81 to 0.81 for *enhance system related communications*. The reliability scores were 0.87, 0.83, 0.82, and 0.89 respectively for each construct. The overall reliability for the 18 item scale was 0.88. The reliability exceeds the suggested cutoff value of 0.70 [14].

Table 4. Factor analysis of fundamental objectives in Phase 2 (n=201)

	F1	F2	F3	F4	Corrected Item-Total Correlation
Maximize standardization, integration and user requirements					
Maximize standardization of system features	0.86				0.78
Maximize functional standardization	0.82				0.77
Maximize system interoperability	0.64				0.67
Maximize automated internal controls	0.61				0.65
Maximize ease of use					
Maximize ease of use		0.85			0.74
Maximize ease of system use		0.77			0.73
Maximize ease of system navigation		0.56			0.59
Maximize system capability					
Enhance explanatory features in the system			0.75		0.72
Enhance geographic location features			0.73		0.69
Enhance e-commerce features			0.62		0.61
Enhance system related communications					
Minimize user interaction with system developers				0.87	0.81
Minimize users' interaction with technical personnel				0.87	0.81
Eigenvalue	5.17	1.62	1.26	1.13	-
% Variance	43.0%	13.5%	10.5%	9.4%	-

Note: loadings greater than 0.3 are reported; the items are grouped by highest factor loading and presented by descending order.

In short, the results obtained in Phase 2 present good reliability and validity measures for both instruments developed (means objectives: 4-factor with 15 items; fundamental objectives: 4-factor with 12 items).

4 Discussion

The findings from our research present an interesting mix of security and usability objectives that any software developer would find useful. The fundamental objectives identified include: *maximize standardization and integration*, *maximize ease of use*, *maximize system capability*, *enhance system related communication*. Typically system developers tend to focus on one or the other set of objectives. For instance past research has typically suggested that perceived ease of use effects perceived usefulness and hence behavioral intention to use [15]. However the measures are not entirely useful to a typical system developer (*viz. c*onstructs such as perceptions of internal control, computer anxiety, playfulness etc). From a security and usability perspective perhaps ease of system navigation and the general perception of easy to use seem more logical.

Another important aspect as always, is related to system related communications. In the literature various proposals have been made. These have ranged from development of *hybrid managers* [16] who can help bridge the gap between technical system developers and actual users to the development of intrinsic competencies for harnessing technology [17]. While all these assertions may present significant theoretical opportunities, typically in organizations we are still to see adequate management of interactions between users and the technical staff. Inability to deal with such relationships results in systems getting abused or not properly used. And thus pose significant security challenges.

The importance of standardization and integration in security and usability cannot be underestimated. A casual review of various security and usability standards itself suggests a plethora of options. In the usability community although ISO standards such as ISO 9241 1995 exist, there is lack of consensus with respect to the conformance methods. Dzida [18] notes, "If a product is claimed to meet a standard, the procedure used in testing the product against the requirements should be specified to guarantee reproducibility of results. Some standards prescribe a certain test method, some recommend a method, and some inform the reader that the procedure used in testing is a matter of negotiation between the parties involved". In information security, while ISO 27799 exists, there are equally other competing standards (*viz.* SSE-CMM among others). Perhaps one of the reason for the inadequacy of existing standards and an existence of a large set is because of a lack of core objectives that need to be achieved in managing security and usability. More often than not the standards seem to be "cobbled" together to fit a purpose. Our research has identified four rather interesting standardization requirements - standardization of features, functional standardization, interoperability and automated internal controls. As a case in point simply consider academic university websites across institutions. Perhaps some functional and feature standardization would come in handy as would access and availability of information. Failure to do so not only makes it difficult to navigate systems but; opens up institutions to several vulnerabilities (e.g. see website breaches at Utkal University India, St Louis University USA among others).

Our research has found that fundamental objectives for security and usability can be achieved if there is a corresponding appreciation of the means to achieve the fundamental objectives. Means objectives identified in this study include: *minimize system interruptions and licensing restrictions*, *maximize information retrieval*, *maximize system aesthetics*, *maximize data quality*.

In our study we found that the higher licensing costs and poor quality of systems and data results in bypassing legal software and many of the controls. This can have serious consequences on the integrity of systems. As a consequence, virus and malware problems are also known to creep in. Grabosky and Smith [19] has argued that proper guardianship helps in preventing such vulnerabilities. Guardians are also known to facilitate usability of system. Retrieval of information from systems has also been a well-researched topic area and sits at the cusp of security and usability dimensions. Griffith and Jakobsson [20] for example note that mother's maiden name, a usual means to retrieve data from financial institutions, can actually be deduced with great accuracy from public records. Some progress has however been made by adding personal knowledge questions for information retrieval, but more so for fallback authentication. From a security and usability perspective enough thought has

not gone into the strategic aspects of information retrieval and their relationship to security and usability. Our research indicates it to be an important objective for consideration.

Another important aspect, as identified in our study, pertains to data quality. In the literature poor data quality has been known to have two implications. First, the security of an enterprise gets compromised (see Redman [21]). This is because security is directly linked to the accuracy of data. Second, usability of the system gets questioned. If the system and the data therein is not useful [22], is out of context, there is typically a loss of ownership. This results in significant security problems.

Theoretical and practical contributions. The major theoretical contribution of the security and usability objectives presented in this paper is their intertwined nature. Typically security and usability have been treated as separate constructs. At best researchers have pondered about security implications of low usability systems or the implications of highly secure systems on lack of usability (see [1, 2, 21, 22]). While both the contentions are worthy of investigation, they fall short of providing a strategic direction for secure and usable system development. We believe that our research provides a theoretical framework for addressing security and usability. The major tenants of our theoretical contribution are:

- Well-grounded security and usability objectives that are based on the values of individuals. Value based objectives are considered much better for strategic planning relative to the alternative based objectives (see [23]).
- Our value proposition combines security and usability. While in the literature calls for aligning the two have been made [2], there has been practically no follow up research. By combining the two constructs we have in many ways presented a well-aligned set of security and usability objectives.

At a practical level, research presented in this paper offers requirement objectives that system developers should use to design security and usability into the systems. Typically security has been considered as an afterthought in the design process [7]. And usability has been addressed in an iterative manner. While system developers seem to develop their own processes in addressing security and usability concerns, a structured framework is however a preferred approach. We believe that the guidance provided by the security and usability objectives described in this paper forms a solid, theoretically grounded, empirically derived basis for the range of development tasks.

5 Conclusion

This paper examines the combination of security and usability of systems in two phases. In Phase 1, we developed value-focused security and usability objectives. A qualitative approach revealed 150 objectives, 91 means objectives and 59 fundamental objectives, grouped respectively into 16 and 8 means and fundamental objectives. A quantitative approach was developed in Phase 2, with the aim of purification, reliability and unidimensionality, from which a parsimonious set of security and usability objectives were derived. 15 means objectives were obtained,

grouped in four constructs, which are: *minimize system interruptions and licensing restrictions*, *maximize information retrieval*, *maximize system aesthetics*, and *maximize data quality*. In terms of fundamental objectives, 12 fundamental objectives were obtained, grouped in four constructs, respectively: *maximize standardization and integration*, *maximize ease of use*, *maximize system capability*, and *enhance system related communication*. We believe that this paper offers a good basis for better understanding of security and usability objectives. Finally, with further research the instruments presented in this paper could be further validated.

References

1. Yee, K.P.: Aligning security and usability. Ieee Security & Privacy 2, 48-55 (2004)
2. DeWitt, A.J., Kuljis, J.: Aligning usability and security: a usability study of Polaris. Proceedings of the second symposium on Usable privacy and security, pp. 1-7. ACM, Pittsburgh, Pennsylvania (2006)
3. Frøkjær, E., Hertzum, M., Hertzum, M., Hornbæk, K.: Measuring usability: are effectiveness, efficiency, and satisfaction really correlated? Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 345-352. ACM, The Hague, The Netherlands (2000)
4. Baskerville, R.: Information systems security design methods: implications for information systems development. Computing Surveys 25, 375-414 (1993)
5. Dhillon, G.: Managing information system security. Macmillan, London (1997)
6. Keeney, R.L.: Value-focused thinking. Harvard University Press, Cambridge, Massachusetts (1992)
7. Dhillon, G., Torkzadeh, G.: Value-focused assessment of information system security in organizations. Information Systems Journal 16, 293-314 (2006)
8. Torkzadeh, G., Dhillon, G.: Measuring factors that influence the success of Internet commerce. Information Systems Research 13, 187-204 (2002)
9. Keeney, R.L.: The value of Internet commerce to the customer. Manage. Sci. 45, 533-542 (1999)
10. Boudreau, M.C., Gefen, D., Straub, D.W.: Validation in information systems research: A state-of-the-art assessment. MIS Quarterly 25, 1-16 (2001)
11. Churchill, G.A.: Paradigm for Developing Better Measures of Marketing Constructs. Journal of Marketing Research 16, 64-73 (1979)
12. Weiss, D.J.: Factor analysis and counseling research. Journal of Counseling Psychology 17, 477-485 (1970)
13. Sharma, S.: Applied Multivariate Techniques. John Wiley & Sons, Inc, New York (1996)
14. Nunnally, J.C.: Psychometric Theory. McGraw-Hill, New York (1978)
15. Venkatesh, V.: Determinants of perceived ease of use: Integrating control, intrinsic motivation, and emotion into the technology acceptance model. Information Systems Research 11, 342-365 (2000)

16. Earls, M.J., Skyrme, D.J.: Hybrid managers — what do we know about them? Information Systems Journal 2, 169-187 (1992)
17. Dhillon, G.: Organizational competence for harnessing IT: A case study. Information & Management 45, 297-303 (2008)
18. Dzida, W.: International usability standards. Acm Computing Surveys 28, 173-175 (1996)
19. Grabosky, P., Smith, R.: Telecommunication fraud in the digital age: The convergence of technologies. In: Wall, D.S. (ed.) Crime and the internet. Routledge, London (2001)
20. Griffith, V., Jakobsson, M.: Messin' with Texas deriving mother's maiden names using public records. In: Ioannidis, J., Keromytis, A., Yung, M. (eds.) Applied Cryptography and Network Security, Proceedings, vol. 3531, pp. 91-103. Springer-Verlag Berlin, Berlin (2005)
21. Redman, T.C.: The impact of poor data quality on the typical enterprise. Communications of the Acm 41, 79-82 (1998)
22. Arts, D.G.T., de Keizer, N.F., Scheffer, G.J.: Defining and improving data quality in medical registries: A literature review, case study, and generic framework. Journal of the American Medical Informatics Association 9, 600-611 (2002)
23. Leon, O.G.: Value-focused thinking versus alternative-focused thinking: Effects on generation of objectives. Organizational Behavior and Human Decision Processes 80, 213-227 (1999)