Heuristics for Evaluating IT Security Management Tools

Pooya Jaferian
University of British Columbia
Vancouver, BC, Canada V6T 1Z4
pooya@ece.ubc.ca

Kirstie Hawkey
Dalhousie University
Halifax, NS, Canada B3H 1W5
hawkey@cs.dal.ca

Andreas Sotirakopoulos
University of British Columbia
Vancouver, BC, Canada V6T 1Z4
andreass@ece.ubc.ca

Konstantin Beznosov
University of British Columbia
Vancouver, BC, Canada V6T 1Z4
beznosov@ece.ubc.ca

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.
ACM 978-1-4503-0268-5/11/05.

Abstract
The usability of IT security management (ITSM) tools is hard to evaluate by regular methods, making heuristic evaluation attractive. However, ITSM occurs within a complex and collaborative context that involves diverse stakeholders; this makes standard usability heuristics difficult to apply. We propose a set of ITSM usability heuristics that are based on activity theory and supported by prior research. We performed a study to compare the use of the ITSM heuristics to Nielsen's heuristics for the evaluation of a commercial identity management system. Our preliminary results show that our new ITSM heuristics performed well in finding usability problems. However, we need to perform the study with more participants and perform more detailed analysis to precisely show the differences in applying the ITSM heuristics as compared to Nielsen's heuristics.

Keywords
Heuristic evaluation, IT security management, computer supported cooperative work

ACM Classification Keywords
H.5.3. Group and Organization Interfaces: Evaluation/Methodology.

Introduction
Information technology security management (ITSM) tools serve several purposes including protection

(network, system, and data protection), detection (tools for threat and vulnerability management), and user management (tools for identity and access management). Evaluating the usability of specific ITSM tools is challenging. Laboratory experiments can have little validity due to the complexity of real-world security problems and the need to situate a specific tool within a larger context and scenario. However, it is difficult to recruit security practitioners (SPs) for interviews, let alone field observations [1]. Direct observation of tool use can be time consuming, as most security work is spontaneous or occurs over many months. As ITSM tool use is intrinsically cooperative, its study inherits the difficulties of studying cooperation. As a result, heuristic evaluation (HE) of ITSM tools could be a viable component of usability evaluations.

The goal of the research presented here is to propose and validate a new set of heuristics for evaluating ITSM tools. The focus of the new heuristics is on finding problems that hinder the use of tools in those ITSM activities that are distributed over time and space, involve collaboration between different stakeholders, and require knowledge to deal with the complexity.

Proposed ITSM Heuristics

Our review of HE literature shows that there are two dominant approaches to developing usability heuristics for a specific software family. On the one hand, researchers extend or adapt Nielsen's usability heuristics for a specific domain (e.g., ambient displays [2]). On the other hand, researchers develop new heuristics based on a specific theory that takes into account the characteristics of the target domain (e.g., heuristics based on the mechanics of collaboration for evaluation of shared visual work surfaces for distance-separated groups [3]).

We chose to develop a new set of heuristics for ITSM rather than extend Nielsen's heuristics. Nielsen's heuristics are based on the theory of action and focus on the dialogue between a single user and the physical world. Therefore, while applying Nielsen's heuristics on ITSM tools may improve the usability of tools by helping users form and work toward immediate goals more effectively; it may not improve the usability of the tool by addressing the socio-cultural and collaborative issues at the level of activity.

To build a set of heuristics for ITSM, we reviewed our previously compiled guidelines for ITSM [4] and their original sources using the theoretical lens provided by activity theory [6]. Activity theory allowed us to interpret the rationale behind each guideline and helped us to consolidate and abstract the guidelines into the following heuristics. Heuristics are viewed as more general yet more powerful than guidelines.

Heuristic 1 - Visibility of activity status: "*Provide users with awareness of the status of the activity distributed over time and space. The status may include the other users involved in the activity, their actions, and distribution of work between them; the rules that govern the activity; tools, information, and materials used in the activity; and the progress toward the activity objective. Provide communication channels for transferring the status of the activity. While providing awareness is crucial, limit the awareness to only what the user needs to know to complete his actions.*"

Heuristic 2 - History of actions and changes on artifacts: "*Allow capturing of the history of actions and*

figure 1- Study protocol overview

table 1- Steps of the study protocol

Consent and Background	We began by obtaining participants' consent and then asking them to complete a background questionnaire through which we obtained demographic information and assessed the background of the participants on HCI and computer security.
Training	We provided training on heuristic evaluation for the participants and then described the set of heuristics that they used for evaluation. Finally, we provided a short introduction on the system that they evaluated.
Evaluation	Participants inspected the interface individually. We provided participants with a list of scenarios and asked them (1) to identify usability problems using the provided set of heuristics; and (2) for each problem, to specify the scenario and the heuristic.
Feedback	After the evaluation, participants were provided with a post-evaluation questionnaire to rate their experience in using heuristics. We then obtained qualitative feedback about experience of participants.

changes on tools and other artifacts such as policies, logs, and communications between users. Provide a means for searching and analyzing historical information."

Heuristic 3 - Flexible representation of information: "*Allow changing the representation of information to suit the target audience and their current task. Support flexible reports. Allow tools to change the representation of their input/output for flexible combination with other tools."*

Heuristic 4 - Rules and constraints: "*Promote rules and constraints on ITSM activities, but provide freedom to choose different paths that respect the constraints. Constraints can be enforced in multiple layers. For example, a tool could constrain the possible actions based on the task, the chosen strategy for performing the task (e.g., the order of performing actions), the social and organizational structure (e.g., number of subjects involved in the task, policies, standards), and the competency of the user."*

Heuristic 5 - Planning and dividing work between users: *"Facilitate dividing work between the users involved in an activity. For routine and pre-determined tasks, allow incorporation of a workflow. For unknown conditions, allow generation of new work plans and incorporation of new users."*

Heuristic 6 - Capturing, sharing, and discovery of knowledge: "*Allow users to capture and store their knowledge. This could be explicit by means of generating documents, web-pages, scripts, and notes or implicit by providing access to a history of their previous actions. Tools should then facilitate sharing such knowledge with other users. Furthermore, tools should facilitate discovery of the required knowledge source. The knowledge source can be an artifact (e.g., document, web-page, script) or a person who possesses the knowledge; if a person, provide means of communicating with them."*

Heuristic 7 - Verification of knowledge: *"For critical ITSM activities, tools should help SPs validate their knowledge about the actions required for performing the activity. Allow users to perform actions in a test environment and validate the results of these actions before applying them to the real system. Allow users to document the required actions in the form of a note or a script. This helps the users and their colleagues to review the required actions before applying them to the system."*

Evaluation of Heuristics

While the proposed heuristics are grounded in empirical data and supported by theory, the effectiveness of the heuristics must be validated in a standard HE process. Our first goal is to compare the effectiveness of ITSM heuristics to that of the Nielsen's heuristics. Our second goal is to investigate the characteristics of an evaluation performed with the ITSM heuristics in comparison to one performed with using Nielsen's heuristics. These characteristics include: (1) the number of evaluators required; (2) background knowledge required; and (3) usefulness, ease of learning, and ease of applying heuristics.

To achieve the aforementioned goals, we performed a between-subjects comparative study of the ITSM heuristics with Nielsen's heuristics. Participants were divided into two groups: those that used Nielsen's heuristics (Nielsen condition) and those that used the ITSM heuristics (ITSM condition). An overview of the study protocol is provided in Figure 1; the details of each step are provided in Table 1.

table 2- Participants' demographics for each condition.

Demographics		ITSM	Nielsen
Group Size (N)		7	7
Gender	Female	6	3
	Male	1	4
Age	Range	25-43	25-42
	Mean	35	32
Level	High school	0	1
	Bachelor	5	2
	Masters	2	3
	Other	0	1
Experience	HCI	3	3.4
	Security (research)	1	0.3
	Security (professional)	0.5	0.5

We chose an Identity Management (IdM) system as the target system for evaluation. An IdM system is used to manage digital identities of users in an enterprise and manage the accesses of those identities to resources. Furthermore, the system allows users to request and approve access to resources and to perform auditing, reporting, and compliance duties. We limited the scope of the evaluation to four typical usage scenarios as the target IdM system had a wide range of functionalities. These scenarios guided the evaluators, who were not domain experts, in performing specific tasks on the IdM system. We piloted our study through several iterations during which we refined the description of heuristics, our study materials, the training process, and the protocol.

Data Analysis

To generate a set of usability problems as the output of the HE in each condition, we needed to first synthesize the usability problems identified by each evaluator and then produce an aggregated list of problems. In order to have a consistent and repeatable methodology of synthesizing the problems, we used the following steps, which were performed by two researchers who resolved any inconsistencies through consensus. In the first step, *problem synthesis*, we decomposed problems that referred to different actions, artifacts, or mechanisms in the interface into their finest level of granularity. This is because each part of a compound problem might have a certain severity and therefore priority for fixing. In addition, if the researchers could not reproduce a problem, it was marked as unknown. We also marked problems as false positives if they were caused by the constraints or requirements of the underlying operating system, hardware/software infrastructure, and business constraints or requirements in the scenario. Then, in the second step, *aggregating problems*, each researcher started with an empty list of aggregated problems. Each usability problem was compared with the problems in the aggregated list. If the problem did not yet exist in the list, it was added. Otherwise, the description of the problem in the aggregated list was refined based on the description of the usability problem. The third step was to *assign severity ratings.* We used two levels of severity: Major and Minor. Three HCI researchers with a background in usable security independently determined the severity of each problem. We then used the median of their severity ratings as the severity for the problem.

Results

In Table 2 we present our participants' demographics in terms of age, gender, and level of education. We also indicate the years of professional and research experience our participants had in HCI and computer security. In order to validate whether the expertise of the two groups was balanced, we calculated scores regarding participants' experience in HCI and computers security, using the weighted average of various experience indicators. Independent sample t-tests revealed no statistical significance between the two groups as measured by a scoring system that incorporated experience and training (HCI experience: Nielsen's-21, ITSM-28; computer security experience: Nielsen's-16, ITSM-13).

table 3- Overview of Identified Problems

Condition	Raw	Known	Major	Minor	FP	Unknown
ITSM	112	82	31	51	9	9
Nielsen	98	68	14	54	23	8
All	210	126	38	88	30	17

table 4- Performance of evaluators in finding usability problems

Condition	Max(%)	Min(%)	Q_1(%)	Q_3(%)	Max/Min	Q_3/ Q_1
ITSM	19.8	2.4	5.2	13.9	8.3	2.7
Nielsen	26.2	4.0	4.8	15.5	6.6	3.2

Table 3 shows the classification of the problems in each condition. The "Raw" column shows the initial number of problems identified by the evaluators. The "Known" column shows the number of problems after synthesis. Of the 126 total known problems, 24 were identified in both conditions.

Effectiveness of heuristics: Based on the above numbers, we compare the effectiveness of the heuristics used in each condition. Since HE is a method that is performed collectively, we calculate effectiveness metrics based on the aggregate of problems from different evaluators. The ultimate criteria for determining the effectiveness of a set of heuristics is whether it finds real problems that a user will encounter in a real work context. However, it is not possible to determine if each usability problem is real or not. The best we can do is to estimate the impact of the potential problem on the users who will use the system. Therefore, we will estimate the effectiveness of our approach based on the following criteria:

Thoroughness: We calculate *thoroughness* as the proportion of the problems identified in each condition. Our results show that evaluation with the ITSM heuristics resulted in finding 65% of total known problems while the evaluation with Nielsen's heuristics resulted in finding 54% of them. To take into account the impact of severity on thoroughness, we use the notion of *weighted thoroughness* by increasing the weight of the major problems. Weighting the major problems as double the minor ones, the weighted thoroughness of ITSM and Nielsen's heuristics are 69% and 50% respectively.

Reliability: It is important for a set of heuristics to be able to identify major usability issues as these may seriously hinder the ability of the user to operate the system effectively and efficiently. We conducted a chi-square test for independence in order to determine whether participants using the set of ITSM heuristics could find more major usability problems for the examined system than the ones using Nielsen's set. The result was statistically significant ($\chi^2(1, 150)=4.46$, p=.035, phi=.19).

Validity: Another aspect of our ITSM heuristics that we evaluated was their ability to yield fewer false positives (FPs) than Nielsen's set of heuristics. Participants who used the ITSM heuristics reported 9 FPs, while participants who used Nielsen's reported 23. Of these, 2 were common issues reported by different participants in both conditions, which left 7 and 21 unique reports of issues marked as FP. The evaluation with the ITSM heuristics yielded significantly fewer FPs than that with Nielsen's heuristics ($\chi^2(1,130)=7.69$, p=.006, phi=-.26).

Performance of evaluators: An important characteristic of a new set of heuristics is the individual performance of evaluators in finding usability problems with them. We show the summary of evaluator performance for each condition in Table 4. In addition to the performance of the strongest and weakest evaluators, we calculated the proportion of problems found by the first and third quartile to eliminate the impact of outliers. We also listed the ratio between the values as an indication of the difference between individual performances.

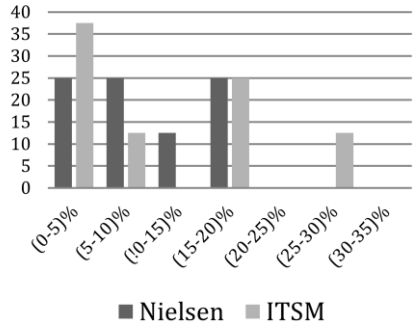

figure 2- Distribution of the proportion of the identified problems in both conditions. The vertical axis shows proportion of evaluators and the horizontal axis shows Proportion of known usability problems.

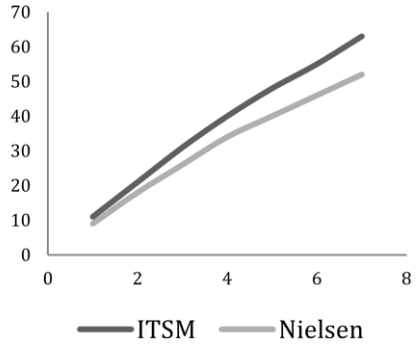

figure 3- Average proportion of problems by aggregate of evaluators. The vertical axis shows averge proportion of problems and the horizontal axis shows Proportion of known usability problems.

These proportions are calculated based on the total problems (126) found. Nielsen et al. [5] observed that the individual differences between evaluators are higher in systems that are more difficult to evaluate. Our results confirm this observation as we found relatively larger individual differences in our two conditions (2.69, 3.25) than in the four experiments reported in [5] (1.4, 1.6, 1.7, 2.2). In figure 2, we show the distribution of the proportion of identified problems by the proportion of the evaluators in each condition. Our results show a different pattern compared to Nielsen et al. [5], but one similar to that seen by Baker et al. [3] who evaluated groupware. To replicate Nielsen's original analysis [5], we formed aggregates of evaluators and found the average number of problems identified by each size of aggregate. Figure 3 shows that adding evaluators will result in a near linear increase in the proportion of identified problems. This is a different trend than reported in Nielsen et al. [5], where the increase rate in the average proportion of usability problems started to diminish when the number of evaluators increased from 5 to 6. In our results, there was little diminishment in the increase rate with 7 evaluators for either condition. This can be attributed to the complexity of the IDM system, as our results are in line with Baker et al [3].

Limitation and Future Work

Our preliminary results show that ITSM heuristics performed well in finding usability problems in the ITSM tools. However, for neither set of heuristics were 7 evaluators enough to achieve saturation in the identified problems. We believe multiple factors contributed to this observation including complexity of the system, the broad evaluation scope (participants visited 20 pages during the evaluation), and our decision to not combine similar problems found in different tasks. To address this problem, we recruited 14 more participants; and we are currently analyzing the results. In addition, we will use the collected data to determine if the evaluators' HCI or computer security background impacts the number and severity of problems. We will also investigate the participants' opinions about the applicability of Nielsen and ITSM heuristics by analyzing the data from the post-evaluation questionnaire, focus groups, and interviews. Finally, we plan to recruit real users of IdM system and ask them to rate the severity of the problems. We will then compare the two sets of heuristics based on those ratings. Finally, as a possible future work, we can identify usability problems in the IdM system in a lab study and compare the results with those of HE.

References

[1] Botta, D., Werlinger, R., Gagn´e, A., Beznosov, K., Iverson, L., Fels, S., and Fisher, B. Towards understanding IT security professionals and their tools. *In SOUPS*. Pittsburgh, PA, 2007, 100–111.

[2] Mankoff, J., Dey, A. K., Hsieh, G., Kientz, J., Lederer, S., and Ames, M. Heuristic evaluation of ambient displays. *In Proc. CHI '03*. 2003, 169–176.

[3] Baker, K., Greenberg, S., Gutwin, C. Empirical development of a heuristic evaluation methodology for shared workspace groupware. *In CSCW*. 2002, 96-105.

[4] Jaferian, P., Botta, D., Raja, F., Hawkey, K., and Beznosov, K. Guidelines for Designing IT Security Management Tools. *In CHIMIT*. 2008, 7:1–7:10.

[5] Nielsen, J. and Molich, R. Heuristic evaluation of user interfaces. *In CHI*, 1990, 249–256.

[6] Kaptelinin, V. and Nardi, B. *Acting with technology: Activity theory and interaction design*. MIT Press,06.