Towards Supporting Users in Assessing the Risk in Privilege Elevation

by

Sara Motiee

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Applied Science

in

THE FACULTY OF GRADUATE STUDIES
(Electrical and Computer Engineering)

The University Of British Columbia
(Vancouver)

October 2011

© Sara Motiee, 2011

Abstract

To better protect users from security incidents, the principle of least privilege (PLP) requires that users and programs be granted the most restrictive set of privileges possible to perform the required tasks. The low-privileged user accounts (LUA) and privilege elevation prompts are two practical implementations of PLP in the main-stream operating systems. However, there is anecdotal evidence suggesting that users do not employ these implementations correctly. Our research goal was to understand users' challenges and behavior in using these mechanisms and improve them so that average users of personal computers can follow the PLP correctly.

For this purpose, we conducted a user study and contextual interviews to investigate the understanding, behavior, and challenges users face when working with user accounts and the privilege elevation prompts (called User Account Control (UAC) prompts) in Windows Vista and 7. We found that 69% of participants did not use and respond correctly to UAC prompts. Also, all our 45 participants used an admin user account, and 91% were not aware of the benefits of low-privileged user accounts or the risks of high-privileged ones. Their knowledge and experience were limited to the restricted rights of low-privileged accounts. Based on our findings, we offered recommendations to improve the UAC and LUA approaches.

Since our study showed that users can benefit from UAC prompts, we investigated the information content for such prompts so that users can assess the risk of privilege elevation more accurately and consequently respond to the prompts correctly. We considered thirteen different information items for including on these prompts mostly based on the results of our first study. Our user study with 48 participants showed that program name, origin, description, digital certification, changes the program applies and the result of program scan by anti-virus are the

most understandable, useful and preferred items for users. To avoid habituation, decrease cognitive load on users and improve users' response to the prompts, we recommend to employ a context-based UAC prompt which presents a subset of information items to users based on the context. A set of guidelines is provided for selecting the appropriate items in different contexts.

Preface

Versions of chapter 3 of this thesis have been published. The author of this thesis performed the users studies presented in this chapter and chapter 4. She also analyzed the data from those studies. She authored the corresponding papers, under the supervision of Dr. Konstantin Beznosov and Dr. Kirstie Hawkey who provided feedback and guidance throughout the research process.

Below are the details of published papers:

- Sara Motiee, Kirstie Hawkey, and Konstantin Beznosov. 2010. Investigating User Account Control Practices. In *Proceedings of the 28th International Conference Extended Abstracts on Human Factors in Computing Systems* (CHI '10). ACM, New York, NY, USA, 4129-4134.

- Sara Motiee, Kirstie Hawkey, and Konstantin Beznosov. 2010. Do windows users follow the principle of least privilege?: Investigating user account control practices. In *Proceedings of the 6th Symposium on Usable Privacy and Security* (SOUPS '10). ACM, New York, NY, USA, 1-12.

- Sara Motiee, Kirstie Hawkey, and Konstantin Beznosov. 2010. The Challenges of Understanding Users' Security-related Knowledge, Behaviour, and Motivations. In *Proceedings of SOUPS Usable Security Experiment Reports (USER) Workshop*. URL: http://cups.cs.cmu.edu/soups/2010/user_papers/Motiee _understanding_user_knowledge_USER2010.pdf

Two user studies were conducted as part of this research. For the first study (explained in chapter 3), we submitted a human ethics application with the BREB number of H09-01702 to UBC Behavioural Research Ethics Board. For the second

study (explained in chapter 4), we submitted an amendment (with the same BREB number) to the first study application. The ethics application and its amendment were approved by UBC Behavioural Research Ethics Board.

Table of Contents

Appendices

List of Tables

List of Figures

Acknowledgments

First, I offer my sincere gratitude to my supervisor, Dr. Konstantin Beznosov, who has supervised and supported me throughout my research.

Special thanks to Dr. Kirstie Hawkey for her mentorship since the initial stages of my research.

Thanks to Dr. Kruchten and Dr. Fels who kindly accepted to be in my committee.

I would like to thank my friends at the Laboratory for Education and Research in Secure Systems Engineering (LERSSE) who provided me with constructive feedback in all phases of my research.

I would also like to thank my dear family; I was away from them for a long time, but they always have had a special place in my heart for their unconditional love.

Chapter 1

Introduction

To limit damages from security breaches, the "principle of least privilege" [27], or PLP for short, requires that each subject in a system be granted the most restrictive set of privileges possible for performing the task at hand.

One practical implementation of PLP in operating systems is a "least-privilege user account" (LUA),[1] which requires users to use accounts with as few privileges as possible for day-to-day work on PCs [32]. To implement this approach, operating system designers have developed various types of user accounts and advise end users to employ low-privileged accounts for their daily tasks [32]. By following this approach, users will be better protected from malware, security attacks, accidental or intentional modifications to system configuration, and unauthorized access to confidential data.

While low-privileged user accounts enhance security, they have not been widely adopted. Indeed, during a Microsoft Financial Analyst Meeting in 2005, it was estimated that 85% of PC users performed their daily tasks using admin accounts [22]. One reason for the lack of LUA popularity is that many simple tasks (e.g., changing the system time when traveling, installing an application) can only be done from an account with administrative privileges ("*admin privileges*" for short) [39]. Running the computer with admin privileges brings convenience for users in terms of performing their daily activities on the computer. However, it also means that

[1] Since LUAs may not necessarily be *least* privileged, we refer to them as "*low*-privilege user accounts".

if malware, spyware, or a virus gets into their machine, it can also install and run with admin privileges. Furthermore, malicious users can more easily gain access to private data with admin privileges. It appears that users often choose the convenience of working with admin privileges over the reduction in risks associated with security breaches.

To alleviate this problem, some operating systems such as Windows Vista, Windows 7 and Ubuntu introduced a new approach for implementing the PLP. In these operating systems, all processes run with non-admin privileges. When a process requires admin privileges for completing its job, it triggers a privilege elevation prompt which asks user' consent for privilege elevation. If user consents to the privilege elevation, the process runs with admin privileges. The developer of these operating systems advise users to think carefully when they see a privilege elevation prompt and make sure what action is about to be performed [43].

Even though there is anecdotal evidence suggesting that main-stream operating systems support users poorly in following the PLP, i.e. users do not employ LUAs and do not respond to privilege elevation prompts correctly, there has been no published empirical data that could inform researchers and practitioners on the actual use of LUAs and privilege elevation prompts by users.

The goal of this research is to investigate how well main-stream operating systems for personal computers support users in following the PLP and what can be done to improve such support.

In the rest of this chapter, we provide an overview of our research in Section 1.1, followed by a summary of our contributions in Section 1.2. Section 1.3 outlines the structure of this thesis.

1.1 Overview

We began our research by investigating users' understanding, behavior and challenges in using current implementations of the PLP in two main operating systems, Windows Vista and Windows 7. We chose these two operating systems as most users use them for their daily computer tasks. In these two operating systems, there are two types of user accounts, protected admin and standard user account. To comply with the PLP, when working with either of these two accounts, user's

processes run with low privileges. If user performs an action that requires admin privileges, a privilege elevation prompt called User Account Control (UAC) prompt is triggered to ask user's consent [2] for privilege elevation. The standard user needs to provide admin credentials in the UAC prompt, however, the admin user only needs to consent to privilege elevation. If users respond to UAC prompts correctly, i.e. consent to legitimate privilege elevation and cancel the prompts triggered by malicious program, the PLP is followed. Windows Vista and 7 developers advise users to use standard account on their computer and respond to UAC prompts carefully. However, it is not clear whether users apply these guidelines. Therefore, it is important to investigate how users utilize each type of user accounts and how they respond to UAC prompts.

To this end, we conducted a laboratory study, followed by contextual interviews with a diverse set of 30 Windows Vista and 15 Windows 7 participants. To maintain ecological validity, we asked all participants to perform study tasks on their Windows laptops. It was perhaps shocking, but not surprising, to find every single participant performing day-to-day activities on own their laptop using an admin account. Most of participants (91%) did not understand the security risks of high-privileged accounts or the benefits of low-privileged ones. In addition to a lack of awareness of security risks, prior experience with the inconvenience of low-privileged user accounts in different contexts of prior discouraged participants from using such accounts. To investigate the use of the UAC approach, we asked participants to complete a set of tasks that raised legitimate and fake UAC prompts to observe their response behavior. Our results show that at least 69% of participants did not use UAC approach correctly. Interestingly, most of these participants (90%) did not have a correct understanding of the purpose of UAC prompts. Also, we found when users are in the context of performing an action, they do not respond to UAC prompt correctly. Based on our findings, we provided recommendations to improve the UAC and LUA. Although we performed the study in Windows Vista and 7, other main operating systems such as Linux and Mac OS X have similar approaches for implementing the PLP.

As a follow up to our first study, we applied one of our recommendations for

[2] We use the term "*consent*" to indicate that the user consents to privilege elevation asked by UAC prompt.

improving the UAC prompts. If users respond to UAC prompts correctly, the PLP can be followed without sacrificing the convenience of working with admin accounts. Also, our study showed that when participants had a partially correct understanding about the purpose of the UAC prompt, they responded correctly to them. They also appreciated the security protection and wanted to receive such prompts.

To improve the UAC prompt, we used one of our first study recommendations which was improving the content of UAC prompt. By focusing on the information content of the prompt, we were able to distinguish the effect of presentation from information content. Our goal was to determine the information items that can assist users in assessing the risk of privilege elevation more accurately so that users can respond to prompts correctly. Our previous study showed us what kind of information is missing from UAC prompt that leads to incorrect response by participants. Based on these findings, we included thirteen different information items on the prompt and conducted a user study to investigate how participants understand and utilize these information items for responding to the prompt. Our user study with 48 participants revealed that most participants understood the purpose of UAC prompt. While many of the items were understood and used correctly by participants, program name, origin, description, digital certification, changes to apply and result of scan by anti virus were the most understandable, useful and preferred items for our participants. We also concluded that it is more beneficial for users to present them a subset of items based on the context. This approach decreases the cognitive load on the users and improves their response accuracy. It also makes the prompt polymorphic which leads to deceasing the habituation. A set of guidelines for selecting appropriate content in each specific context is provided in this thesis.

1.2 Contributions

Here we provide a summary of our contributions in each of the two user studies that we performed:

1. **Investigating user account control practices:** The first contribution of this study was to reveal the users' understanding, behavior and challenges in us-

ing practical implementations of PLP in Windows Vista and 7. Our study showed, although 71% of participants had a partial understanding of the limitations and rights of admin and non-admin user account types, 91% of participants were not aware of the security risks of high-privileged accounts or the security benefits of low-privileged ones. All used admin accounts and were not motivated to use non-admin account on their own computers because of unawareness about these accounts benefits and the limitations they had faced using these accounts.

We also found that 69% of our participants did not employ the UAC approach correctly as they either disabled it or consented to any UAC prompt, especially the prompts that are raised when they were in the context of doing an action or initiated an action themselves. These participants had an incorrect understanding of UAC, developed an incorrect response rationale and responded to prompts incorrectly.

Another contribution of our study was to propose a set of recommendation for improving the UAC and LUA approaches. We recommend conveying the purpose and benefits of these approaches to users, raising UAC prompts in fewer situations, providing relevant information on the prompt for assisting users in responding to prompts, integrating UAC functionality with other security software, providing users with default low-privileged accounts, and making the use of low-privileged account convenient in order to ensure that users continue to use them.

2. **Information Content for Assessing the Risk in Privilege Elevation:** Our contribution in this part of our research was identifying the information content for the UAC prompt so that users can assess the risk of privilege elevation more accurately. For this purpose, we evaluated how participants understood and utilized thirteen different information items for assessing the risk of privilege elevation and responding to UAC prompts in different contexts. Our results showed that the program name, origin, description, digital certification, changes to apply and result of scan by anti virus were the most understandable, usable and preferred items for our participants. The other contribution was to propose a set of guidelines for developing a context-

based UAC prompt. Such a prompt presents a subset of information items to users based on the specific context and decrease their cognitive load for responding the prompt.

It should be noted that the variability of our participants with respect to age, gender, educational level, background, and security knowledge and expertise is a major strength of our two user studies.

1.3 Outline

The remainder of this thesis is organized as follows.

1. Chapter 2 provides the related work and background information for this thesis. It includes the related work on the principle of least privilege. It also explains and compares the approach of three main-stream operating systems for implementing this principle. Since one approach of these operating systems for implementing the PLP is based on warnings and we have focused on this kind of warning in our research, this chapter presents the related work on warning literature and recommendations for designing warnings.

2. Chapter 3 presents our first user study which investigates users' understanding, behavior and challenges in using the PLP implementations in Windows Vista and 7.

3. Chapter 4 presents the information content that we considered for UAC prompt for supporting users in assessing the risk of privilege elevation and the user study for evaluating the effectiveness of this information content.

4. Chapter 5 summarizes the contributions of this thesis, and introduces directions for the future research.

Chapter 2

Background and Related Work

The aim of our research is to investigate how well main-stream operating systems support users in following the PLP and how the technology can be modified for improving this support. There are different mechanisms for implementing the PLP. We categorize these mechanisms into those implemented by operating systems and other mechanisms external to operating systems.

We first explain and compare the mechanisms of four main-stream operating systems, Windows Vista, Windows 7, Mac OS X and Ubuntu for implementing the PLP in section 2.1. Then, we discuss the related work on the mechanisms external to operating systems in section 2.2.

The current mechanism of all above operating systems for enforcing the PLP is based on raising warning messages for elevating the privileges of user. Since we have focused on this kind of warning message in our research, the related work on warning messages and recommendations for warning design are also discussed in this chapter.

2.1 Operating systems implementations of principle of least privilege

The operating systems implement the PLP by offering various types of user accounts with different privileges. Most of operating systems include a high privileged (admin) account and a low privileged (normal) account. The admin account

has the privileges to install programs and change the system settings; however, normal account does not have such privileges.

This design of main-stream operating systems suffers from the limitation that every program has the same privileges as the account under which it has been launched, whether the user wants this or not. This limitation has been exploited by malware performing operations unintended by users. To address this limitation, Windows Vista, Windows 7 and Ubuntu introduced a new approach for enforcing the PLP. In these operating systems, all processes run with low privileges. When a process wants to perform an operation that needs admin privileges, the process privilege should be elevated. For this purpose, in Windows Vista and 7, a privilege elevation prompt is triggered by operating system to ask user's consent for privilege elevation. In Ubuntu, this can be achieved by privilege elevation prompt or running a command by user. Mac OS X uses a similar approach; however, this is not the default behavior.

In the following sections, the user account model and privilege elevation mechanism of four main-stream operating systems are explained and compared with each other. Table 2.1 and table 2.2 summarizes the features of user account model and the privilege elevation prompt in these operating systems. Also, table 2.3 lists the actions that need admin privileges and trigger the privilege elevation prompt in each of these operating systems.

2.1.1 Windows Vista and Windows 7

Early Microsoft Windows operating systems did not have the concept of different user accounts on the same machine. In Windows NT and later versions of Windows, there are two types of user accounts: admin and normal (standard). In Windows 2000, XP Professional, and Server 2003, there is also a "power user" account type that has more permissions than a normal account, but does not have some admin permissions.

Microsoft advises users to use low-privileged user accounts, or LUA for short, for their daily computer use and recommends that admin and power user accounts only be used by trustworthy and knowledgeable users [32]. However, in all versions of Windows, all user accounts are created as admin by default; and users

	Windows Vista / 7	**Ubuntu**	**Mac OS X**
Account types	Protected admin Standard Guest	Root Admin Desktop Custom	Admin Standard Parental control Sharing only Group
Default account	Protected admin	Admin	Admin
Switching between accounts	Available	Available	Available
Recommended account to create	Standard	Not available	Not available
Options for creating accounts	Protected admin Standard	Desktop user but the privileges can be modified	Admin Standard Parental control Sharing only Group

Table 2.1: User account model of main-stream operating systems

	Windows Vista	**Windows 7**	**Ubuntu**	**Mac OS X**
Name	UAC prompt	UAC prompt	Authentication dialogue	Authentication dialogue
Response options	Ok Cancel	Yes No	Authenticate Cancel	Ok Cancel
Input Admin	No input	No input	Admin password	Admin password
Non-admin	Admin password	Admin password	Admin password	Admin username & password
Tuning	Enabled Disabled	Four levels	NA	Can be enabled for Admin
Prompt Types	4 types	4 types	Wording of some prompts differ slightly	Wording of some prompts differ slightly

Table 2.2: Privilege elevation prompt of main-stream operating systems

Task Description	WV	W7	Ubuntu	Mac OS X	
				Admin	Standard
Install a program	✓	✓	✓		✓
Install / uninstall a device driver	✓	✓	✓		✓
Install drivers downloaded from Windows Update or included in the operating system	✓		NA	NA	NA
Install an ActiveX control	✓	✓	NA	NA	NA
Install updates	✓		✓	✓	✓
Copy or move files into system directory	✓		✓	NA	NA
View/change system-wide settings	✓		✓		✓
Modify security settings with the Security Policy Editor snap-in	✓	✓	NA	NA	NA
Open or change the Firewall settings	✓		✓		✓
Configure remote desktop access	✓	✓		NA	NA
Configure parental controls	✓		NA	✓	✓
Add or remove a user account	✓		✓	✓	✓
Change privilege elevation prompt settings	✓	✓	NA	NA	NA
Change a user account type	✓		✓	✓	✓
Browse another user's directory	✓			NA	NA
Configure Automatic Updates	✓		✓		
Backup and restore Files	✓				✓
Running Disk Defragmenter	✓		NA	NA	NA
Pair bluetooth devices to the computer	✓				

Table 2.3: Tasks that trigger a privilege elevation prompt

continue to use admin accounts on their systems. Moreover, using non-admin accounts inconveniences users as many simple tasks (e.g., changing the system time) could only be done with an admin account [39].

To make the use of LUAs convenient for users, a user account control (UAC) mechanism was introduced in Windows Vista and revised in Windows 7. UAC has a goal of allowing all users, including local administrators, to run with non-admin privileges when admin privileges are not required. Microsoft has mentioned in [38] that the UAC approach is designed to help prevent malware from installing without the user's knowledge, using "bundling" and social engineering, browser exploits,

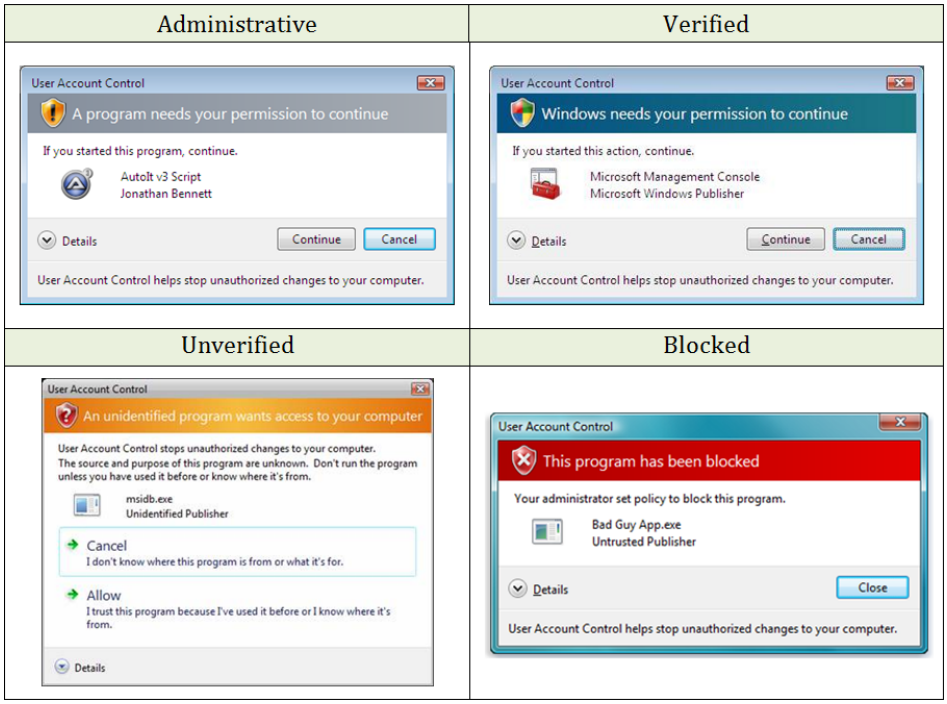


Figure 2.1: Sample of UAC prompts for admin user in Windows Vista

and network worms.

In Windows Vista and 7, there are two types of user accounts, protected admin and standard user account. With a standard user account, users are not allowed to install programs, change system settings, and perform other tasks that require admin privileges. During the Windows Vista and 7 installation process, the user is prompted for a user account information. By default, an admin account is created. But Microsoft advises users to create a standard account after operating system installation for their daily usage. When a standard user account attempts to perform a task that requires admin privileges, a UAC prompt is triggered which asks for the password of an admin account. When a protected admin attempts to perform a task that requires admin privileges, a UAC prompt is triggered which asks user to consent to the privilege elevation. Figure 2.1 - figure 2.4 show the samples of UAC prompts in Windows Vista and 7 for admin and standard users. Windows Vista and 7 developers recommend users to think carefully when they respond to a UAC

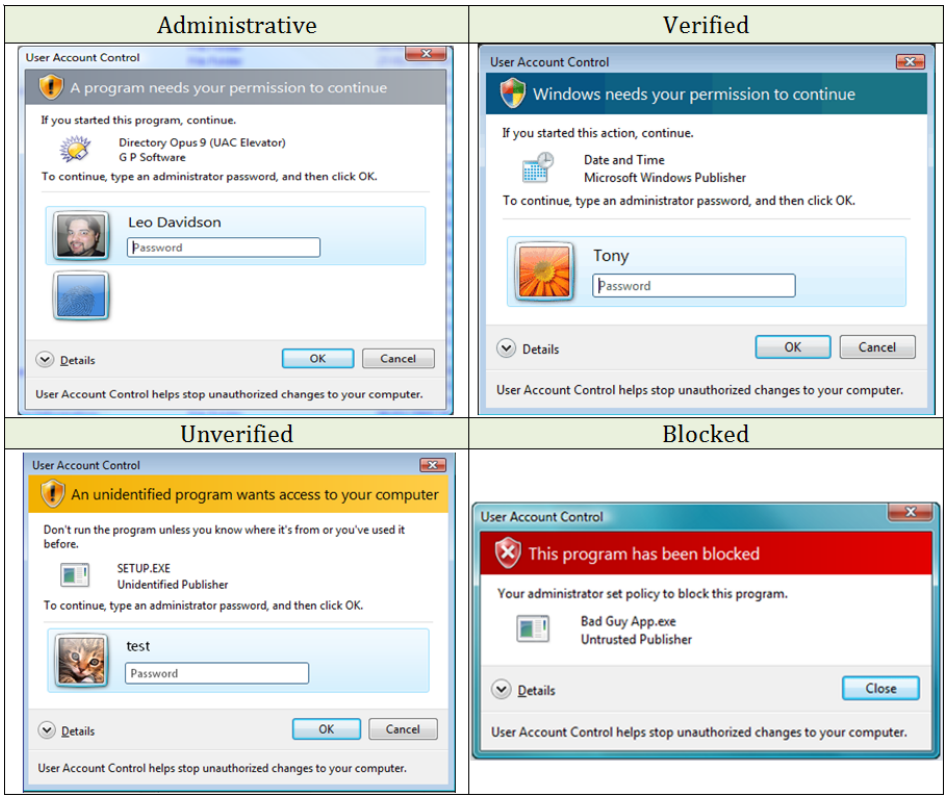


Figure 2.2: Sample of UAC prompts for standard user in Windows Vista

prompt and to make everyone, even administrators, enter passwords in the prompt; so that they take advantage of UAC features for the security of their system [43].

The underlying UAC approach in Windows Vista and Windows 7 is the same; however, Windows 7 has reduced the number of UAC prompts. The tasks [25, 39, 40] that raise UAC prompts are listed in Table 2.3. Windows 7, by default, prompts the user when a non-Windows executable asks for privilege elevation [26]. Therefore, when the user changes Windows settings, she is not prompted, but when non-Windows applications (e.g. installing a new software) request administrative changes, a UAC prompt appears. We note that omitting the prompts and privilege elevation without asking users are in contrast to the main goal of UAC: preventing silent installation of malware. The effectiveness of this tradeoff has yet to be evaluated.

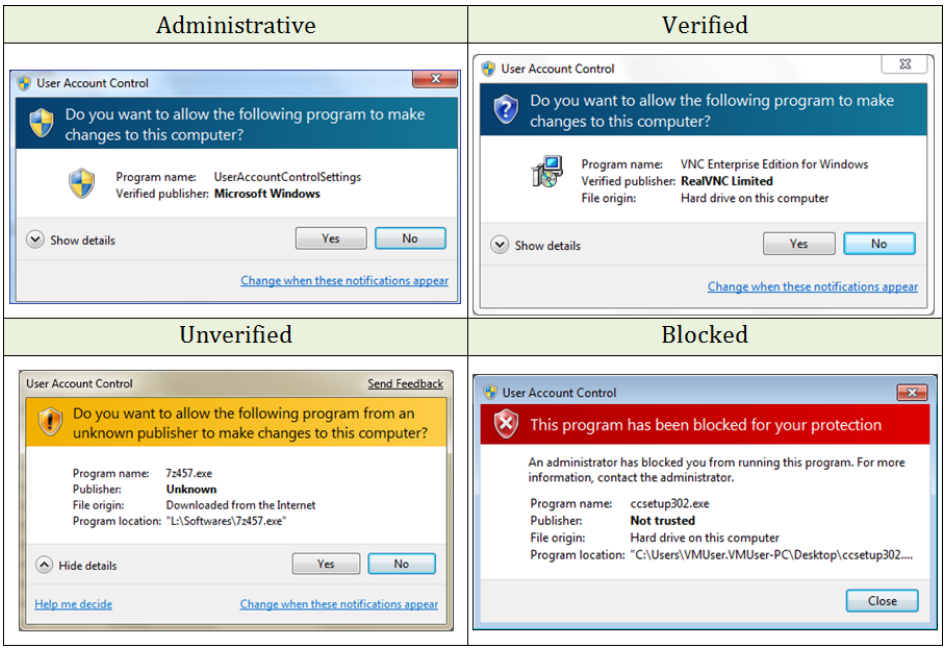


Figure 2.3: Sample of UAC prompts for admin user in Windows 7

Both Windows Vista and 7 implement four types of UAC prompts, color coded to inform users of the potential security risk of installing or running an application or applying a change. The prompt type is based on the executable's publisher. Table 2.4 lists the UAC prompt types (labeled as they are referred in this thesis) and their color schemes in both operating systems; there are some differences.

In Windows Vista, users can only disable and enable the UAC prompts and prompts always appear on a *secure desktop* in which the screen is dimmed. However, in Windows 7, user can adjust the UAC behavior in four modes:

1. Always notify: This is the default mode in Vista.

2. Only notify when programs make changes to the computer (Default): In this mode, the user is only prompted when a non-Windows executable asks for elevation.

3. Do not dim the desktop: The difference between this mode and the default mode is that prompts happen on the user's desktop rather than on the secure

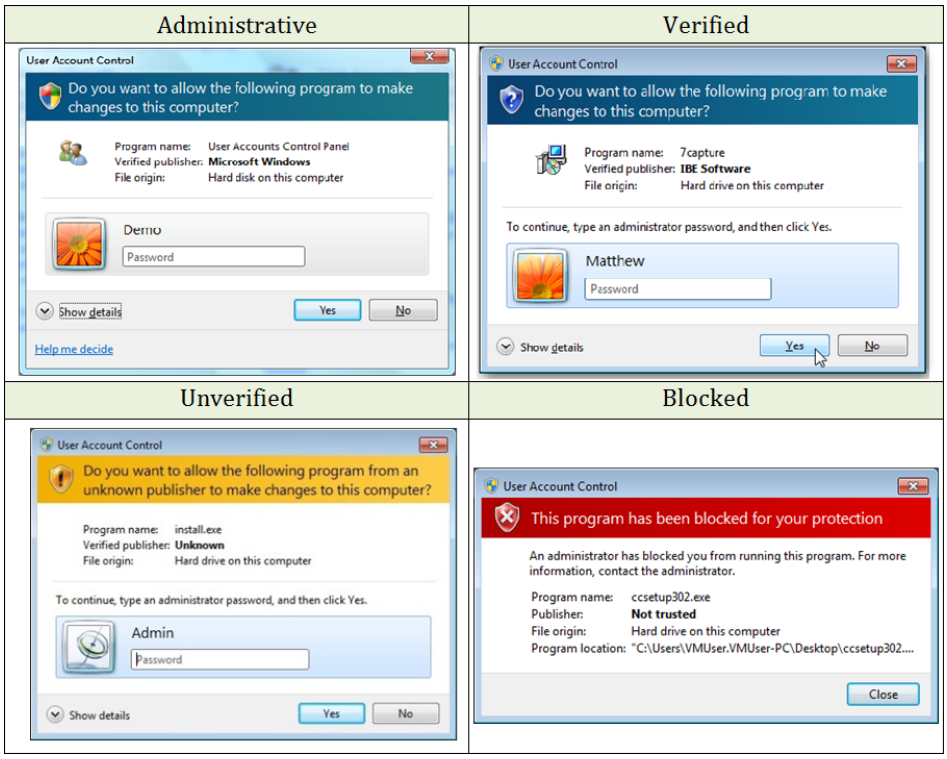


Figure 2.4: Sample of UAC prompts for standard user in Windows 7

desktop.

4. Never notify: This turns UAC prompts off.

UAC is complementary to LUA; that is, users can employ one, both, or neither of the two. When UAC is enabled, the type of user account is not critical for following the PLP. In this case, if the user responds to UAC prompts correctly, she follows the PLP; if she does not respond correctly, the PLP is violated. However, if UAC is disabled, the type of user account determines whether the user follows or violates the PLP.

Improvements to UAC prompt

There is anecdotal evidence that users find UAC prompts annoying as these prompts are confusing and are triggered in many situations. Due to users' complaints, Nor-

Label used in the thesis	Window Vista		Windows 7		App. Type Whose Action Causes a Prompt
	BG	Shield	BG	Shield	
Blocked	Red	Red	Red	Red	Blocked publisher or blocked by Group Policy
Administrative	Blue/ Green	Gold	Blue	Blue/ Gold	A Windows Vista/7 administrative App.
Verified	Grey	Gold	Blue	Blue	Authenticode signed & trusted by the local computer
Unverified	Yellow	Red	Yellow	Yellow	(Un)signed and or signed but not trusted by the local computer

Table 2.4: UAC prompts color coding

Figure 2.5: Sample of Norton UAC prompt

ton and Security Stronghold have improved UAC prompts. The Norton UAC which is shown in figure 2.5 is in Beta mode and has three new features in comparison to Windows UAC prompt [20]:

1. Don't ask me again: The Norton UAC has an option called "Don't ask me again" which user can check when a UAC prompt is triggered. If an application is previously granted access in a specific context and user checks "Don't ask me again", the next time user runs this application in the same context, no UAC prompt is triggered. Although Norton does not define "context" specifically, it has provided the following example. "There is a difference

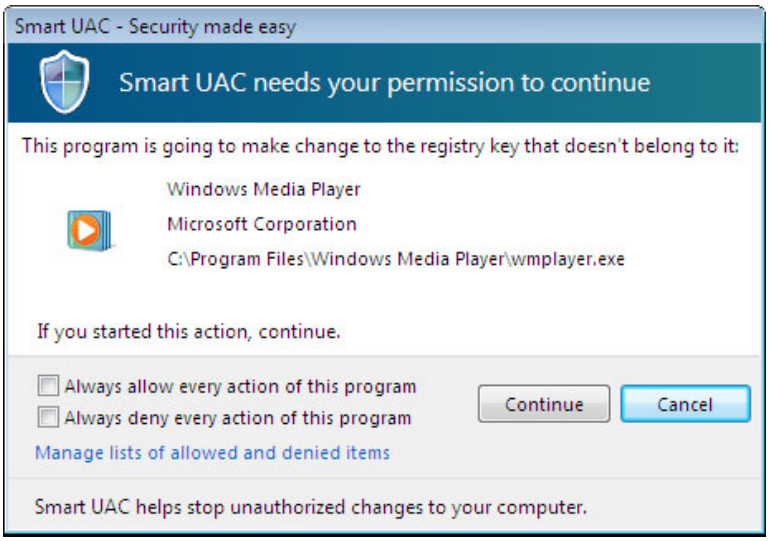


Figure 2.6: Sample of Smart UAC prompt

between regedit.exe launched from the run box in the start menu, regedit.exe originating from a shortcut double-click, and regedit.exe launched from a double click on a .reg file (and the context actually changes with each .reg file), and regedit.exe launched by an application."

2. It shows whether the application is digitally signed (and if so, who signed it) and whether the application resides in a protected directory. However, it is not defined what the protected directory means.

3. White list and black list: Norton builds a white list and blacklist of applications to reduce the number of prompts. For this purpose, each time a user responds to a Norton UAC prompt, the Norton Labs UAC Replacement sends meta information (what caused the prompt, and the response information) to their server. Norton builds the white and black lists by aggregating these data.

The other improvement to Windows UAC prompt is Smart UAC [30] shown in figure 2.6. Its new features are as follows:

1. Showing the admin action: The Smart UAC shows which admin action has triggered the prompt.

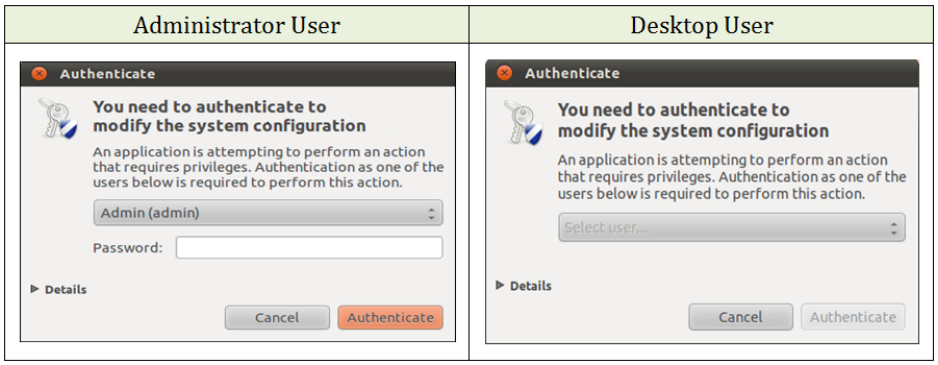


Figure 2.7: Sample of authentication dialogue for admin and desktop user in Ubuntu

2. Always allow / Always deny: In the Smart UAC the user can select the "Always allow" or "Always deny" options so that the current action or all the actions of the current program will not trigger a prompt in the future.

3. Scanning the program: Smart UAC scans the program with large data base of 400,000 threats and notifies the user if the program is malicious.

The usability of neither Smart UAC nor Norton UAC have been evaluated, so it is not clear how their improvements have changed users' behavior.

2.1.2 Ubuntu

Among Linux distribution, Ubuntu is currently the most popular distribution as it receives more than 2,200 hits per day on the Distrowatch site while Mint, the second contender, receives 1,900 hits per day [21]. Also, it uses the PolicyKit component for controlling the system-wide privileges in the operating system. The UNIX distributions that use PolicyKit component (such as RedHat and Fedora) implement the PLP differently. In these systems, in contrast to earlier UNIX systems, non-privileged processes communicate with privileged ones in an organized way. This component ensures that the root access is not granted to an entire process and provides a finer level of privilege management.

There are four user account types in Ubuntu: root (super user), admin, desktop user and custom. The root account is disabled by default. It means that users cannot

login directly with root or use the `su` command to become the root user. The root account can be enabled which is not advised. The admin account is the default account and has fewer privileges than root, but can perform most of admin tasks. This is the first account which is created in the system. The subsequent accounts are created as desktop user which does not have admin privileges. If the privileges of desktop or admin user are modified, the account type becomes custom.

If an admin, desktop or custom user needs to do a task that requires admin privileges such as application installation or system configuration, she should provide the user name and password of an admin account before performing the task. Table 2.3 shows the list of actions that need admin privileges in Ubuntu. If the user is using the command line interface, she can use the `sudo` command and enter the admin password to elevate her privileges. When using sudo, the password is stored by default for 15 minutes. Therefore, the admin actions that are performed in the next 15 mintunes, do not require password. If the user is using the graphical user interface, a privilege elevation prompt, called "Authentication Dialogue" is triggered which asks for credential of an admin account [24]. Figure 2.7 shows a sample of authentication dialogue for admin and desktop user. Authentication dialogues are the same however, the wording of some of them slightly differs. For example when installing a program, the message on the prompt is: "To install or remove software, you need to authenticate", while when changing firewall settings, the message is "You need to authenticate to modify the system configuration". The authenticate dialogue cannot be disabled, but it is possible to add a user to sudeors for specific tasks. In this case, if user executes sudo for those tasks, no admin password is requested.

2.1.3 Mac OS X

In Mac OS X, there are five different user account types: Admin, standard, parental control, sharing only (guest) and group (collection of different accounts).

The admin account is the default account created during the operating system installation or activation. The admin account can do the admin tasks, however, the user needs to provide the admin password in an "Authentication Dialogue" before performing some of these tasks. While Apple advises that this account be reserved

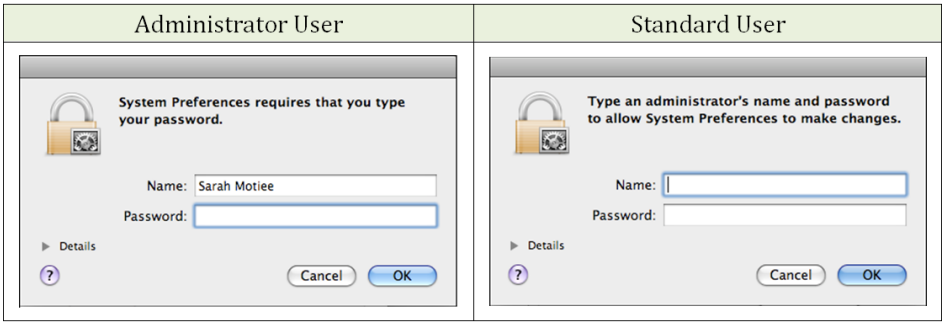


Figure 2.8: Sample of authentication dialogue for admin and standard user in Mac OS X

for making changes to the system and installing system-wide applications [15], it is the only account created during the OS installation. Also, the user has an option of configuring her machine to log into this account automatically (i.e., without entering a password).

The standard account cannot perform admin tasks by default. However, if the standard user wants to do an admin task, she should provide the user name and password of an admin account before performing the operation in an authentication dialogue. Table2.3 shows the tasks that trigger authenticate dialogue for the standard account and admin account. It is possible to configure the admin account to provide password before each admin task. Figure 2.8 shows a sample of authenticate dialogue for admin and standard account in Mac OS X. All the authenticate dialogues are the same and the only difference is the name of the program which triggers the prompt.

2.1.4 Comparison and conclusion

All the four operating systems provide low and high privileged user accounts. However, there are some differences between the privileges of admin accounts in each operating system. Also, all the four operating systems provide a privilege elevation prompt for elevating the privileges of the user. There are some differences in the wording, appearance and the tasks that trigger the privilege elevation prompt.

In Windows both admin and standard user can do admin tasks but the admin should consent to privilege elevation via UAC prompt and standard user should

provide the admin credentials in the UAC prompt. In Mac OS X, the admin can do most admin tasks without providing consent or credentials in the authentication dialogue except for few tasks in which the user should provide the admin password before performing the task. However, the standard user needs to provide the admin credentials for performing every admin task in the authentication dialogue. In Ubunto, both the admin and the desktop user should provide the password of an admin account for every admin task in the authentication dialogue.

Therefore, the approach of all these operating system for implementing the PLP is based on providing user accounts with high and low privileges and getting the user consent before elevating the privileges via a privilege elevation prompt.

2.2 Other implementations of principle of least privilege

Other mechanisms external to the operating system have been developed for applying the PLP. One approach is Sand-boxing [17], which provides a tightly-controlled set of resources for a program to run. However, the rules for specifying resources are static and adding privileges to a running program is difficult. iOS, the operating system of Apple devices (iphone, ipod, iPad and AppleTV), uses the sand-boxing approach. To improve sand-boxing, Watson et al. [44] designed Capsicum for UNIX which introduces new security primitives to support compartmentalization. Compartmentalization is the decomposition of monolithic application code to components that runs in independent sandboxes. To take benefit of security features of Capsicum, application developers should adapt their applications to use Capsicum primitives. Wruster et al. [51][50] have also presented a new approach to prohibit applications from modifying each other files. Their system is called Configd and is implemented in UNIX. However, it has not been shown how this approach prohibits an attacker from modifying system files. Another approach for applying the PLP is SELinux [29] which is a feature in Linux and allows system admin to determine the resources that each application or user can access in the system. Every application, user and resource has a label and system admin should define the policies using these labels. A similar feature to SELinux is AppArmor [37]. It is a security module for Linux kernel and is developed by a team sponsored by Novell. By using AppArmor, each application can have a profile that specifies the actions

that the program can take. System admin should create these profiles. AppArmor developers claim that AppArmor is easier to use than SELinux. It also has a learning mode in which the operations of application that are not mentioned in the profile, are added to the profile instead of being prevented. The new profile will be the correct profile of application. Both AppArmor and SELinux are not intended for average user.

Another approach for implementing the PLP is asking the user to confirm the permissions for an application when it is started or during the run time. Some Java Web Start applications follow this approach [13]. Schneider [28] has also proposed a new approach for enforcing the PLP, but has not implemented or evaluated his approach. His approach consists of a reference monitor which intercepts all program actions and based on privileges held by the issuer of the action, blocks those that would be disruptive. Moreover, two packages have been developed for Microsoft Windows for applying the PLP. The first, CapDesk [34], is a distributed file browser and application launcher that was developed to reduce the threat of viruses and trojan horses for everyday users of the Web. It allows users to browse files and open them with the associated application; opening a file in CapDesk launches a caplet, which only has the authority to edit the file that was double-clicked. A security evaluation found the approach to have merit, but no user evaluation was conducted. The second, Polaris [35] was developed by HP Labs for Windows XP; its design was based on CapDesk. Polaris launches each application without the authority to access any user files. When a user opens a file via double clicking or the file chooser dialog box, Polaris grants the application access to that file. There were plans to install Polaris on consumer PCs that HP ships, but the current status of these plans are unclear. Also, a usability study of Polaris showed it has usability problems that prohibit users from making correct security decisions.

2.3 Usability of implementations of principle of least privilege

Except the usability study of Polaris, we are unaware of any study directly evaluating the usability of technologies implementing the PLP; however, there has been some work looking at user account models for shared home computers. Egelman et

al. [9] presented and evaluated a new user account model called Family Accounts, which provides a shared family account as well as personal accounts. Switching between accounts happens quickly and does not close running applications. Sharing information is easier using this model and users can switch accounts only when they require personalization or privacy. However, this model does not encourage the use of low-privileged accounts.

2.4 Security warnings

As mentioned in section 2.1, triggering a security warning (privilege elevation prompt) is part of Windows Vista, Windows 7, Ubuntu and Mac OS X mechanism for applying the PLP. We are unaware of any related work investigating the effectiveness of warnings that aim to prevent users from installing and running malware on their system in general or privilege elevation prompt in particular. However, there are studies that have proposed and evaluated other security warnings.

Prior research in warning literature suggests that warning messages should be used as the last solution for reducing a risk [45]. Also, warnings should communicate the risk and clear instructions for avoiding the risk [33]. In addition to studies on computer warnings, different studies have proposed and evaluated security warnings. They also have provided recommendations for designing security warnings. Table 2.5 lists the recommendations provided in the usable security literature for designing security warnings.

Zurko et al. [54] have evaluated the usability of Lotus Notes security alters that aim to prohibit users from running unsigned active content. They performed the study in a 500-person organization and found that 59% of participants allowed the unsigned content to run. They suggest educating the users or including more information in security-related interfaces.

Egelman et al. [10] evaluated the effectiveness of active and passive phishing warnings in current web browsers. Their finding suggest that active warnings are more effective than passive warning and using passive warnings is the same as using no warning. In their study, of the participants who saw the active warnings, 79% chose to close the phishing web sites. The authors offered recommendations for improving phishing indicators. They suggest a phishing indicator should inter-

Number	Recommendation	Reference
1	Reduce the number of prompts	[23] [10] [7] [5] [36]
2	Communicate the risk and threat level	[23] [10] [7] [5] [52]
3	Be jargon free	[23] [5] [12] [10]
4	Recommend an action	[23] [10] [52]
5	Be easy to understand	[23] , [5]
6	Be short	[5] [12]
7	Interrogate the user about the context and provide guidance on how to proceed	[4] [42]
8	Educate the users or include more information in the warning	[54] [23]
9	Prevent habituation	[10][5]
10	Interrupt the current task	[10]
11	Fail safely if user ignores the warning	[10]
12	Provide clear choices	[10]
13	Avoid warning with the same response options	[4]
14	Increase user effort for response	[4]
15	Randomize the order of response options in a list	[4]
16	Delay the display of final confirm option	[4]
17	Use polymorphic messages	[4]
18	Inform clearly about the consequences of actions	[3]
19	Inform about the probability of risk occurrence	[3]

Table 2.5: Recommendations for designing security warnings

rupt the user's primary task, prevent habituation, provide clear choices, alter the look and feel of phishing sites, and fail safely if the user ignores or misunderstand it.

Sunshine et al. [36] studied users' behavior in responding to SSL warnings in web browsers by running a survey of 400 Internet users. They found participants' risk perception is correlated with their decisions to obey or ignore the SSL warning. They also designed two new SSL warnings and conducted a laboratory study with 100 participants to compare their new warnings with three existing SSL warnings.

They found that a large number of subjects ignored SSL warnings when using Firefox v2 (90%), v3 (55%), and Internet Explorer v7 (90%). The new warnings had an overall better effectiveness (45% and 60% of subjects ignored the first and second warning respectively). Based on the results, they suggested improving the design of warnings using appropriate colors and text, and decreasing their frequency.

Brustoloni et al. [4] implemented a polymorphic warning to assist users in deciding about opening a risky email. Their system asks a series of questions from the user and user chooses one of the answer options. The order of answer options changes every time to avoid habituation. Based on user's responses, the risk of opening email is classified as justified or unjustified. They have evaluated their approach using a user study and found polymorphic messages help participants make better security decisions.

Wogalter [46] proposed the Communication-Human Information Processing Model (C-HIP) to analyze and identify the reasons that a particular warning is ineffective. In this model, a communication is sent to a human receiver to trigger a behavior. The behavior depends on communication impediments, communication processing, and personal variables of the receiver. Cranor [5] enhanced this model to consider the human in the loop in secure systems. Five communication types are defined in the framework: warnings, notices, status indicators, training, and policies. The privilege elevation approach of Windows Vista, Windows 7, Ubuntu and Mac OS X is communicated via warnings while LUA is not communicated to users. The use of LUA is only encouraged in online documentation of these operating systems.

Chapter 3

Investigating User Account Control Practices

3.1 Introduction

The goal of this chapter is to investigate how well main-stream operating systems for personal computers support users in following the principle of least privilege, (or PLP for short), and what can be done to improve such support. Our particular objectives are to determine (1) how well users are aided by the technology to follow this principle, (2) what challenges they face, (3) what factors motivate their behavior, and (4) what are the areas of potential failure for current PLP mechanisms.

We narrowed the scope of our research to Windows Vista and Windows 7 because these two main operating systems are used by most of the users [41]. Also, their approach for implementing the PLP is almost the same as the approach of other main-stream operating systems such as Mac OS X and Ubuntu. The more recent generation of operating systems, e.g. iOS, uses the sand-boxing approach for implementing the PLP in Apple electronic devices such as iPod, iPad and iPhone. However, it is mandatory for application developers to define static rules for specifying the required resources and it is difficult to add privileges to a running program. Also, Apple advises users to run only signed applications on their devices. It is not clear whether users follow the Apple guidelines to benefit from iOS security features.

In addition to low privileged user accounts (or LUA for short), Windows Vista introduced user account control (UAC) [39], which was intended to make the use of LUAs more convenient and therefore reduce incentives for violating the PLP. As explained in chapter 2, with UAC, all users, including local administrators, can work with non-admin privileges when such privileges are not necessary. A UAC prompt is raised when one of the user's processes requires admin privileges (e.g., when installing software or changing system settings). UAC was revised in Windows 7 to reduce the number of prompts by default and to allow users to customize which prompts they receive. If UAC is disabled,[1] the type of user account determines whether the PLP is followed (in case of a non-admin account) or not (in case of an admin account). However, if UAC is enabled on a user's system, it is not critical what type of user account is in use; as long as the user responds to UAC prompts correctly, the PLP is followed. Given this interdependency between LUA and UAC and the critical role of the two in the support for the PLP, we studied the behavior of users in employing LUA as well as in responding to UAC prompts.

To this end, we conducted a laboratory study, followed by contextual interviews. We recruited 30 Windows Vista ("*WV*") and 15 Windows 7 ("*W7*") users in order to observe any changes in their behavior according to the different UAC implementations on these Windows platforms. None of the demographics of our WV and W7 participants were statistically significantly different, except for the years of experience with the operating system. The participants had a wide range of educational levels (from high school to Masters) and the 16 (out of 45) non-student participants had a variety of occupations. It was perhaps shocking, but not surprising, to find every single participant performing day-to-day activities on own their laptop using an admin account. To better understand the factors affecting the use of LUA approach, we asked the study participants to complete a user account creation task, and we probed their knowledge about LUA. Although most created an appropriate low-privileged user account in the study task, participants were not motivated to employ a low-privileged account for their own daily computer usage. Furthermore, 91% of participants did not understand the security risks of high-privileged accounts or the benefits of low-privileged ones. In addition to

[1] UAC prompts can be disabled by the user.

a lack of awareness of security risks, prior experience with the inconvenience of low-privileged user accounts in different contexts of prior discouraged participants from using such accounts.

To investigate the use of the UAC approach, we asked participants to complete a set of tasks that raised legitimate and fake UAC prompts to observe their response behavior. Our results showed that at least 69% of participants did not use UAC approach correctly. These were participants who either consented[2] to a fake random, i.e., not correlated with their current action, UAC prompt (49%) or disabled UAC (20%). Interestingly, most of these participants (90%) did not have a correct understanding of the purpose of UAC prompts. On the other hand, those participants who had a partial understanding of UAC did not consent to the fake random prompt. It was not, however, the case for another fake prompt that was triggered as the result of participants' action during installation: all but 2 participants consented to both the fake and real UAC prompts raised during this task. This result suggests that when users initiate an action that might require admin privileges, they do not respond correctly to the subsequent UAC prompts.

Based on our findings, we offer several recommendations to improve UAC and LUA approaches. Operating system developers should communicate the purpose and benefits of both UAC and LUA to users through the interface itself, rather than only through the technical documentation from the OS vendor. Furthermore, either users should be made aware of the the distinction between UAC and other security-related mechanisms (e.g., personal firewall, anti-virus software), or UAC should be integrated with the other mechanisms. Furthermore, UAC prompts should be raised consistently, in selective and limited situations so that users do not ignore them due to habituation. These prompts should communicate enough information about their purpose and the risks they intend to mitigate so that users can respond correctly to them. To improve LUA, in addition to the admin account created upon installation of the OS, users should be provided with an initial, default, low-privileged user account and be encouraged to use it for their daily work. However, to ensure users continue following LUA, users must be able to conveniently apply modifications on their PCs from within that low-privileged account.

[2]We use the term "*consent*" to indicate that user consents to privilege elevation asked by UAC prompt.

In the remainder of this chapter, we first describe the methodology of our study and its results in Sections 3.2 and 3.3, respectively. Section 3.4 provides a discussion of our results, as well as recommendations for applying the PLP. We discuss the limitations of our study in Section 3.5 and conclude the chapter in Section 3.6.

3.2 Methodology

As we designed our methodology, we referred to Cranor's "human in the loop" framework [5] for analyzing the human factors associated with secure systems. This allowed us to ensure that we observed and considered the various factors that might impact the success of the communication mechanisms of the UAC and LUA approaches (e.g., the prompts in UAC). We aimed to answer the following questions in regards to UAC and LUA:

1. Do users notice the communication mechanism of the UAC and LUA approaches?
2. Do users comprehend and appropriately apply the UAC and LUA approaches?
3. How do users' personal variables, capabilities, and intentions impact their behavior in employing UAC and LUA approaches?

We employed a laboratory study, followed by a contextual interview. This multi-method approach allowed us to mitigate the biases of any one approach and increase the methodological strengths [16]. Security is not usually the primary task or goal of users, therefore, a user study methodology needs to be carefully considered [8]. Because users respond to UAC prompts and manage user accounts infrequently and irregularly, it would be difficult to observe their behavior during normal computer use. We therefore chose to expose users to a set of predefined and controlled tasks, including those that would raise UAC prompts, so that we could gather observational data about their behavior. Furthermore, because participants may not be motivated to apply security practices when using study data and equipment, we had them conduct the experimental tasks on their personal computers. This allowed us to observe them in an environment similar to their normal usage context. We targeted laptop users so that sessions could be held at the university.

3.2.1 User study protocol

We used the same protocol for WV and W7 participants. After signing the consent form, each participant completed the background questionnaire, which had questions about their computer usage pattern, such as usage hours, experience, and used operating systems. This questionnaire is shown in Appendix A.1. Then participants installed software (provided on a USB disk) to record their voice and capture their laptop's screen. We also recorded their screen using a video camera, as the recording software did not capture the UAC prompts raised on the dimmed screen. We observed participants as they were completing the tasks and asked them to think aloud. There were two main parts to the study. The first was designed to investigate the knowledge, behavior and motivations of participants in using UAC approach. The objective of the second part was to learn about participants' account usage behavior and their knowledge about the LUA approach. We did this last, so as not to prime participants on the purpose of the study during the first part. At the end of the study, participants uninstalled the applications that they installed on their laptop during the study.

Part 1: Examination of UAC practices

We asked participants to perform three tasks on their laptops. These tasks were designed to raise two different types of UAC prompts (Verified and Unverified). To increase the ecological validity of the study, we did not provide detailed instructions for performing the tasks. Instead, we presented participants with three hypothetical task scenarios and asked them to perform the same steps that they would normally take. They were told that task completion was not the goal and that they could refuse doing the task if they did not perform such an activity during their normal computer usage. The instructions for doing the task is shown in Appendix A.2. The tasks were as follows:

1. T1: Getting an application for playing a DVD. We presented participants with different options (such as downloading free software, buying software online or from a store, getting application from a friend) and asked what approach they usually took. If they usually downloaded and installed software, they were asked to perform the same steps in the study session. We

observed their decision process for downloading and installing an application, including their response to the UAC prompts and other warnings and messages.

2. T2: Receiving the installation file of a text editor application on a USB disk from a friend who recommended installing the application. Participants were asked whether they installed the application in such a situation. If they responded "yes", they were requested to install the application as they would in a similar real life situation. Installation of this application raised an Unverified UAC prompt.

3. T3: Downloading and installing a specific spyware remover application, recommended by a security expert. This installation triggered a Verified UAC prompt.

While performing these tasks, participants were prompted with two additional fake UAC prompts. The first fake prompt was raised by an application which was wrapped in the screen recorder application installation file. This application was installed on the participant's computer without her notice and raised an Unverified UAC prompt named "UpdateCache" three minutes after the screen recorder installation finished; we explicitly chose a name that was unrelated to their current tasks. Since this prompt was not correlated with the participants' actions, we call it the "Fake Random" prompt (FR for short). Participants faced the FR prompt while they were doing one of the study tasks. While Figure 3.1 shows the average timeline of UAC prompts during the user study, the interleaving of FR with the other UAC prompts raised depends on the speed with which tasks were completed. We observed participants' response to this unexpected UAC prompt.

The second fake prompt was shown during the installation of the text editor. When the installation file ran, the first Unverified UAC prompt was a fake one with a name similar to the application and the second prompt was the real one (also an Unverified UAC prompt). Since this fake prompt was correlated with the installation task, we call it the "Fake Installation" (FI) prompt. We observed participants' response to prompts that appeared during installation.

After performing the tasks, we replayed to each participant the video capture of their screen and interviewed them about their understanding of the tasks and

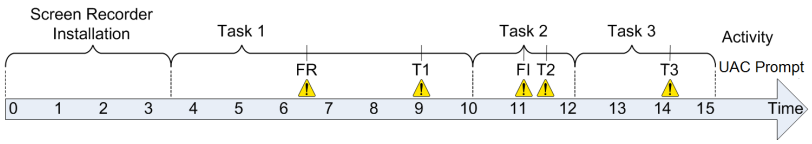


Figure 3.1: Typical timeline of user study tasks and corresponding UAC prompts, FR: Fake Random prompt, T1: Task 1 prompt, FI: Fake Installation prompt, T2: Task 2 prompt, T3: Task 3 prompt, Time scale: Minute

their rationale for the actions they took. In particular, they were asked about their knowledge of the UAC prompt, its interference with their computer usage, its different types, their rationale for responding to these prompts, and their reasoning for responding to fake prompts. Conducting the interview in the context of user study tasks helped participants to understand the questions better and remember their prior experiences more easily. We also contrasted participants' answers to interview questions with their behavior in the user study tasks to decrease the self-report issues. The interview questions are shown in Appendix A.3.

Part 2: Examination of LUA practices

Participants were first asked about the differences between admin and standard user accounts. They were then presented with a scenario in which they were asked to create a user account for their brother who wanted to use their laptop for some tasks such as email, browsing, and using Microsoft Office. By giving them this task, we observed their familiarity with user account management and their decision making processes for account creation. We then probed each participant about their rationale for creating the account in the user study task, the account they used on their own computer and their reasons for its usage, their experience with other user account types, and the challenges they face when using them. For WV participants, we also asked them about the UAC prompt they faced before creating the account to determine whether they were aware of the difference between it and those they received during the first part of the study. This prompt was not raised when the default UAC settings were used in W7. As all W7 participants used the default settings, none received this prompt and we did not ask them about it. The

Property	WV		W7		Total	
	N = 30	%	N = 15	%	N = 45	%
Gender (F / M)	13 / 17	43 / 57	6 / 9	40 / 60	19 / 26	42 / 58
Student (Y / N)	18 / 12	60 / 40	11 / 4	73 / 27	29 / 16	64 / 36
Technical background (Y / N)	10 / 20	33 / 67	8 / 7	53 / 47	18 / 27	40 / 60
Primary OS - Vista or 7 (Y / N)	27 / 3	90 / 10	12 / 3	80 / 20	39 / 6	87 / 13
	Mean	Range	Mean	Range	Mean	Range
Age (Years)	26.3	18 - 50	23.6	19 - 30	25.4	18 - 50
Daily computer usage (Hours)	6.9	1 - 15	7.7	3 - 14	7.2	1 - 15
Computer experience (Years)	11.7	2 - 27	10.5	1 - 23	11.3	1 - 27
Daily WV or W7 usage (Hours)	5.2	0.3 - 12	5.3	2 - 10	5.2	0.3 - 12
WV or W7 experience (Years)	1.5	0.3 - 3	0.3	0.1 - 1	1.1	0.1 - 3

Table 3.1: Participants' demographics

interview questions are shown in Appendix A.3.

3.2.2 Participants

We recruited 30 participants for Windows Vista ("*WV*") and 15 participants for Windows 7 ("*W7*") from both the university and general community. We sent out messages to email lists of several UBC departments, posted messages to Craigslist and Kijiji, and pinned flyers to community bulletin boards. During recruitment, we asked respondents their age, gender, degree, major and occupation to ensure a diverse population for our study. Accordingly, we selectively sampled from the pool of responses in order to achieve breadth in the characteristics of our participants. All participants were paid $10 CAD for their participation. Table 3.1 shows the demographics of our participants. It is challenging to recruit a sample that represents the real users' population, however, our sample included a diverse set of participants. Our participants had a wide range of educational levels (from high

Expert. Level	WV		W7		Total	
	N=30	%	N=15	%	N=45	%
Low	7	23	2	13	9	20
Medium	16	53.3	8	53.3	24	53.3
High	7	23.3	5	33.3	12	26.7

Table 3.2: Participants' computer expertise

school to Masters) and the 16 non-student participants had a variety of occupations such as teachers, secretaries, managers, and photographers.

Also, our participants had different levels of computer knowledge and expertise. To show this, we assessed the participants' computer experience by asking them to indicate how difficult they found performing the following six tasks: copying and moving files, installing software, searching on Internet, installing an operating system, administering a network server, and programming. We categorized their computer expertise as low, medium, or high, as shown in Table 3.2. We also refined our categorization based on participants' performance during the downloading and installation tasks in the study.

We also assessed how cautious participants were about the security of their systems by asking about the frequency of updating their anti-virus software, frequency of full system scans and the security software they have on their computers. We categorized participants' level of security cautious as low (WV: 37%, W7: 33%), medium (WV: 50%, W7: 47%) or high (WV: 13%, W7: 20%).

The properties of our WV and W7 participants were not statistically significant different, except for the years of experience with the operating system being studied; the W7 participants were early adopters of Windows 7, which had only been released for a few months at the time of the study.

3.2.3 Analysis

To analyze the data, we used a card sorting approach [19]. We did not use qualitative methods such as grounded theory because our objective was to categorize the participants' behavior and report the results descriptively. We also aimed to conduct quantitative analysis on the data to extract the correlation between different aspects of participants' behavior.

To analyze the data using the card sorting approach, participants' responses to the interview questions were written on index cards. The index cards for each question were then sorted into multiple piles so that cards representing similar responses were in the same pile. We associated a theme with each pile, that represented participants' knowledge, behavior, and motives based on the corresponding question. The sorting and naming of the piles was done iteratively to find participants' behavioral patterns.

3.3 Results

In the following two sections we present results from the first (UAC practices) and second (LUA practices) parts of user study, respectively.

3.3.1 UAC practices

In this section, we report our participants' knowledge and opinion about UAC prompts, their responses to UAC prompts during the user study, as well as their reported rationale for responding to these prompts during their normal computer usage. We contrast their actual behavior to their reported rationales to determine any mismatches and the underlying reasons for their behaviors.

When comparing the responses and behaviors of WV and W7 participants, we used the χ^2 test; when its assumptions were not met (i.e., cells had an expected count <5), we used the Fisher's exact test. Since in most cases, there was no statistically significant difference between WV and W7 participants, we report the overall behavior of all 45 participants unless such a difference exists.

We found that at least 69% of our participants did not employ the UAC approach correctly. These were participants who disabled UAC (20%) or consented to FR (49%). The latter did not have a correct understanding of the purpose of UAC prompts. All participants but two consented to both the fake (FI) and real prompts triggered during the second installation task; when they initiated an action, they were unlikely to respond to the subsequent prompts correctly.

Knowledge type		WV(N=30)		W7(N=15)		Total(N=45)	
		N	%	N	%	N	%
Terminology		1	3	5	33	6	13
Recognition		29	96	15	100	44	97
Purpose	Getting user's permission	4	13	3	20	7	15
	User initiated operation	4	13	3	20	7	15
Difference between Verified and Unverified prompts		3	10	3	20	6	13
Difference between administrative and other prompts		6	20	N/A	N/A	6	20
Operations raising prompt	Installing application	4	13	5	33	9	20
	Installing application and changing settings	4	13	0	0	4	8
	Privilege elevation	0	0	1	6	1	2
	Installation plus incorrect answers	11	36	4	26	15	33
	Did not know why raised	11	36	5	33	16	35

Table 3.3: Participants' knowledge about UAC prompts.

Knowledge of UAC prompts

The responses of our participants to interview questions indicated that none fully understood the UAC approach. Table 3.3 shows the knowledge our participants had about different aspects of UAC: knowing the terminology; recognizing the prompts; partially understanding the purpose of UAC prompts; understanding the difference between Verified, Unverified and Administrative prompts; and knowing the operations that trigger prompts. All participants recognized the UAC prompts, except for one WV participant whose family member had disabled UAC during her laptop setup. The only significant difference between WV and W7 participants was that more W7 participants knew the term UAC ($p = 0.012$, Fisher's exact test).

Based on our participants' explanations about the purpose of UAC prompts, we categorized them as having a partially correct understanding (PartiallyCorrect) or incorrect understanding (Incorrect). We classified 30% of participants (third row of table 3.3) in the PartiallyCorrect group; they perceived UAC as a mechanism for getting users' permission before applying any change to the system or ensuring that the user has started the operation.

Most participants did not understand the difference between various UAC prompts

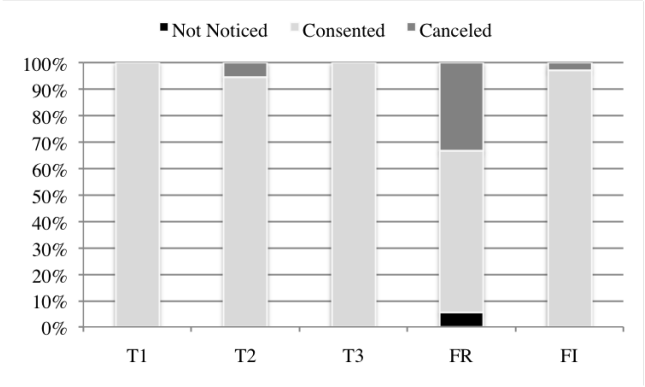


Figure 3.2: Percentage of generated prompts that were not noticed, consented to, and canceled

types. Only 13% of participants perceived the Unverified prompt as being potentially more dangerous, a possible virus, or an unknown application for the computer. Only 20% of WV participants associated the administrative prompt with an administrative task, Windows related operations, or a configuration change prompt versus application related prompt. As all W7 participants used the default settings for UAC, they did not receive the administrative prompt.

Response to UAC prompts

Some participants did not receive all the potentially raised UAC prompts in the study. Nine participants (WV:6, WV:3) had previously disabled UAC on their laptop. Furthermore, not all participants completed all of the user study tasks. One, who was very cautious about downloading and installing, canceled T1 in the middle and did not start T3. Another, who did not regularly download and install applications, did not do T1 and T3. One other participant needed the name of the application to download, so she did not do T1. One, who had disabled UAC did not do T2 because she only installed software that she had heard of before. Three did not do T3 because they were concerned about the compatibility of the spyware remover with their anti-virus software. There were also two participants that did not respond to a prompt which was raised. These participants did not notice the generated FR prompt. When a UAC prompt is not triggered as the result of user's

Reason For Disabling	Total	
	PartiallyCorrect	Incorrect
All prompts ask the same thing	3	1
Interfering with computer troubleshooting	2	0
Did not know the reason, as a family member disabled	0	2
Getting "Java Update" prompt in each startup	1	0

Table 3.4: Number of participants who disabled UAC and their reasons

Reported rationale for responding to UAC prompts in daily tasks	Response to FR in user study , N=34			
	PartiallyCorrect		Incorrect	
	Consent	Cancel	Consent	Cancel
I always confirm without reading	0	0	10	0
If I initiated action raising prompt, I confirm; otherwise I decide after reading	0	10	3	1
If I initiated action raising prompt, I confirm; otherwise I cancel	0	0	5	0
I read & decide based on familiarity with the program	0	0	3	0
I read & decide based on the relevance of prompt with my current action	0	1	0	0
I always cancel Unverified prompts & confirm others	0	0	0	1

Table 3.5: Rationale for responding to UAC prompts and response to FR

action, it is minimized in the task bar and may not be noticed.

Figure 3.2 shows from the prompts that were generated in the study, what portion were not noticed, consented to, or canceled. Since all participants used admin account on their laptop and did not change the default settings of UAC, they did not require to provide admin credentials on the prompts.

As mentioned before, 9 participants had disabled UAC. Table 3.4 indicates their reasons for doing so. We asked the remaining participants their rationales for responding to UAC prompts during their daily usage of computers. Below we contrast participants' reported rationales and their actual responses to the fake prompts of the user study.

Situation	N=34	
	Consent	Cancel
Performing no action	4	1
During downloading, installing or running an application	11	3
Waiting for an action to complete	4	2
Browsing	3	6

Table 3.6: Situation in which FR was received

Fake random prompt (FR)

Of those who received FR, 61% (49% of all participants) consented to this fake prompt. Six percent did not notice it, as they were interacting with another application when the prompt was raised, and 33% canceled the prompt. Table 3.5 lists the participants' response to FR and their claimed rationale for responding to UAC prompts during daily computer usage. The response of all PartiallyCorrect participants matched their reported rationale for responding to UAC; they all canceled FR, although one canceled the prompt without reading it.

Only two Incorrect participants canceled FR; the rest confirmed it. The participants who stated they always confirm UAC prompts behaved as they reported. Most of those (except one) who said they read or cancel prompts that are not triggered as the result of their actions, consented to FR. Also, those who claimed they read the prompt and decided their action based on their familiarity with the program, consented to FR.

We asked participants about their reason for consenting to the FR. While three did not know the reason and four just consented to every prompt, 14 participants thought this prompt was related to the task that they were doing. Also, one participant consented to FR to update his system.

We also investigated when our participants received the FR. Table 3.6 shows the situation in which our participants received this prompt. This table and the previous qualitative results show when participants are in the context of downloading, installing or running an application, they do not respond to UAC prompts correctly.

Expected number of prompts	WV	W7	Total
Did not know	2	1	3
One prompt	12	9	21
Two or more prompts	8	1	9

Table 3.7: Participants' expectations of the number of prompts to be raised when installing an application

Reason For Confirming	WV	W7	Total
Did not know	6	2	8
Continue the installation	2	6	8
Do not notice the difference between two prompts	3	0	3
The first click was not received by computer	1	0	1
Always confirm	0	1	1

Table 3.8: Participants' reasons for confirming FI

Fake installation prompt (FI)

The second fake prompt was FI which was raised during the installation of the text editor application. When the installation file ran, the first UAC prompt was a fake one with a name similar to the application and the second prompt was the real one. Of the 35 participants who viewed the FI prompt (9 disabled UAC and one system did not generate the FI), only 2 participants did not consent. One checked the details of the prompts and, since he got two, canceled the installation; he had stated that he consented to all the prompts. The other did not allow FI since he always cancels Unverified prompts. Therefore, most participants did not respond correctly to the UAC prompts when they initiated an action that triggered prompts.

We asked the participants who consented to FI how many prompts they expect to receive when installing an application. As shown in Table 3.7, 3 did not know how many prompts they should get and consented to FI to continue the installation, and 9 consented to FI since they expected to receive 2 or more prompts during single installation. Although 21 participants expected to receive one prompt, they consented to two consecutive prompts; their reasons for confirming FI are shown in Table 3.8.

		WV		W7		Total	
		N=30	%	N=15	%	N=45	%
Annoying	PartiallyCorrect	2	25	2	33	4	28
	Incorrect	13	59	6	66	19	61
	All	15	50	8	53	23	51
Prefer to Disable	PartiallyCorrect	1	12	2	33	3	21
	Incorrect	11	50	5	55	16	51
	All	12	40	7	46	19	42

Table 3.9: Number of participants who found UAC annoying and preferred to disable it

Opinion about UAC prompts

We asked participants whether they found UAC annoying, and whether they would disable it or not (Table 3.9). Most PartiallyCorrect participants did not find UAC annoying. They appreciated giving permission before changes were made to the system and being informed if someone tries to install something on their system. Only 21% preferred to disable UAC; they were confident that their expertise, their use of security software, and their performance of regular back-ups kept their systems secure.

On the other hand, more than half of the Incorrect participants, found the prompts annoying and preferred to disable them. They gave several reasons, such as having an awareness about their own actions, having a lack of awareness about the purpose of prompts, interference with troubleshooting of their PC, UAC having the same functionality as anti-virus, the time consuming activity of responding to prompts, and a preference for automating the UAC functionality by operating system. The rest of the Incorrect participants did not complain about UAC because they treated it as a part of the procedure of doing an action (e.g. installation or configuration), or they believed the prompts are beneficial because of protection they offer through monitoring the correctness of their actions. However, these participants did not take advantage of this security mechanism as all allowed the fake prompts.

Difference between Windows versions

We asked the 14 W7 participants who had experienced Windows Vista UAC prompts about the difference between UAC prompts in Windows Vista and 7. Of the 5 PartiallyCorrect participants, 4 noticed the decrease in the number of prompts and 1 other appreciated the ability to tune the settings. The Incorrect participants did not recognize any change, thought the number of prompts had increased, or did not remember the prompts in Windows Vista. Four participants had disabled UAC in Windows Vista; 3 of them did the same in Windows 7, and 1 changed the settings so that the screen was not dimmed.

3.3.2 LUA practices

In this section we present participants' knowledge about the types of user accounts, their rationales for using various account types on their system, their prior experience with user account creation, the result of the user account creation task, their experiences with using low-privileged user accounts, and the challenges they face in using such accounts. The results reported in this section include both Windows Vista and Windows 7 participants as both operating systems have the same user account types.

All of our participants used an admin account, and most were not aware of the security risk of high-privileged user accounts or the benefits of low-privileged ones. Many created a standard account in the user study task, understanding the sufficiency of such account for daily tasks and having concerns about the unwanted changes that can be made by an admin account. However, none were motivated enough to use a standard account for their own daily usage.

Knowledge about user account types

Our participants' knowledge about the differences between user account types was limited to the capabilities and rights of each account type. Most did not show any understanding of the security risks and benefits.

When we asked participants about the difference between admin and non-admin accounts, thirteen did not know the difference; the others mentioned various differences such as admin being able to modify the system (26), manage other

User Account Details		N=45
Number of user accounts	One	39
	Two	5
	Four	1
Guest Account	Enabled	0
	Disabled	45
Main user account	Password protected admin	36
	Admin without password	9

Table 3.10: Number of participants with various user account settings on their laptops

users' rights (9), and have more control on the computer (6). Moreover, two said that if an application is installed by the admin account, non admin accounts cannot access it.

We also asked whether participants were aware of any security risk associated with using an admin account; most (36) were not aware of any risk. Of the rest, 5 were aware of the possibility of applying inappropriate changes by themselves when using admin accounts; and 4 were aware of the feasibility of unwanted and unknown changes by a malicious user. However, both groups preferred to use an admin account for convenience, choosing to keep their system secure by performing regular backups and using security software.

User account usage experience

Most participants did not have any experience with non-admin accounts on their own computer, however, most had experienced them in public and workplace settings. Table 3.10 shows the current user account settings on participants' laptops. All participants used the default admin account, whether or not they installed the OS themselves. We probed the 32 participants who knew their account type for their reasoning for using an admin account. Three participants did not know why (3); the others mentioned different reasons, such as having complete access to change everything (17), being unaware of any benefits of using an account of another type(10), owning the computer (6), being the only user of computer (5), having a need to log on and log off if using a non-admin account (5), admin being the default option (3), and being unaware of non-admin accounts (1).

Non-admin account usage		N=45
Did not know		12
Not used		5
Only used on home computer		5
Used Guest	Public computer	10
Used Standard	Family computer	4
	Work computer	7
	School	2

Table 3.11: Participants' experience with using non-admin accounts (not on their current personal computer)

We also asked participants about the type of user accounts they have used on their own previous home computers. Except for five participants, all used admin accounts or did not remember their account type. Two of these five participants were Linux users who used a non-admin user account to avoid applying wrong and accidental changes on their systems. Two others quit using non-admin accounts due to the inability to install applications. The fifth was a developer who created a standard account to test developed software.

Most participants did not use a non-admin account on their own computer. However, they have used it on public or workplace computers (Table 3.11). These participants either complained about the inability to install software (8) or were satisfied because they used the account for only a few tasks (8). Some (3) of those who used a non admin account in the workplace preferred to have fewer privileges so that IT-admin can control things; however, they preferred to use an admin account on their home computer.

User account creation task

Table 3.12 shows the results of the user account creation task. Most participants appropriately created a low-privileged (standard or guest) user account in the user study task. They mentioned different reasons such as guest or standard accounts being sufficient for browsing the Internet and using email and Microsoft office (19), avoiding any application installation or unwanted changes on their systems (17), avoiding giving the same level of control as their own to somebody else (4), using the default option (2), preserving their privacy (5), not needing more than

User Account Creation		N=45
Familiarity with the procedure	Familiar	36
	Unfamiliar	4
	Partially familiar	5
Created user account type	Standard	32
	Guest	8
	Admin	1
	Not done	4

Table 3.12: Number of participants who created different user account types in the user study task

one admin account (2), and not being able to create a second admin account (1).

Therefore, 19 (42%) participants understood that daily computer activities do not require any high-privileged user account and 17 (38%) participants knew that in contrast to a high-privileged account, a low-privileged account cannot apply unauthorized changes on their system. Despite such understandings, none of our participants used a non-admin account on their own laptops.

We asked participants about the situations in which they would create an admin account. Fourteen did not know whether they would ever do so.

Seventeen indicated that they would create such an account for a person who has an appropriate level of knowledge (4), wants to share the computer with them (4), needs to install software or perform an administrative task (2) or is trusted (5). Also, two mentioned the both last two reasons.

Fourteen participants would not create an admin account for anybody because of concerns about incorrect changes to their systems (4), having control on their computers themselves (5), and being concerned about the competence of the new user (1). Two thought that their system could not have two admin accounts; and two preferred to share their account instead of creating a new one.

Therefore, 23 (51%) participants understood that admin account should be used by knowledgeable and trusted people for administrative tasks, and a high-privileged user account may apply undesired modifications to their system. However, all these participants used an admin account on their own laptops.

User Account Type	System	User	N
Not Done	-	-	21
Provide details for admin in start up	Home computer	Personal	7
Admin	Home computer	Family member	3
	Work computer	Colleague	3
Non-admin	Home computer	Family member	5
	Home computer	Personal	3
	Work computer	Colleague	2
	Work computer	Personal	1
Guest	Home computer	Family	3

Table 3.13: Prior user account creation experience

Prior user account creation experience

We asked participants about their prior experience with user account creation (Table 3.13). Half the participants had such an experience, and all but one applied a correct rationale when selecting the user account type. This participant created an admin account because he was unaware of the non-admin account type.

Six participants had created admin accounts for family members or colleagues. This was either to share the computer (5) or due to their lack of awareness of non-admin accounts (1). Five participants created non-admin accounts on their own home computer for family members for various reasons such as preserving their privacy, avoiding unwanted changes, and the limited requirements of their family. Two created non-admin accounts on their office computers so that colleague could perform a few tasks (e.g., printing). One participant, a developer, created a standard account to test some software. Only three participants had created non-admin accounts for their own usage. While one quit using a non-admin account due to the inability to install programs, the others (Linux users) still used it.

3.4 Discussion

Because the UAC approach and low-privileged user accounts rely on users making security decisions, users should be supported in making such decisions. Our analysis and discussion reveal how effective the UAC and LUA approaches were in communicating security-related actions to participants, and whether participants were able to comprehend these communications and respond to them correctly. We

also discuss how participants' personal variables and motivation influenced their behaviors in using these security mechanisms. Finally, we discuss how well the PLP is followed by participants using the LUA and UAC approaches.

3.4.1 User account control

We found that at least 69% of participants did not employ the UAC approach correctly because 20% disabled the UAC prompts, while using admin accounts, and 49% consented to a fake random prompt.

Communication delivery

The UAC approach is communicated to users by prompts. To be successful in communication delivery, users should notice UAC prompts and pay attention to them in order to process the prompts. UAC prompts were effective in capturing our participants' attention as all but two participants noticed the prompts raised during the user study. However, many participants did not carefully consider the UAC prompts and respond to them correctly when they were in the context of installing or running an application, especially when they had initiated the action themselves. Moreover, they were not aware of any risk that consenting to a UAC prompt may create.

Comprehension and application

The majority (70%) of participants had an incorrect understanding of the purpose of the prompts. This incorrect understanding left their laptops vulnerable to security breaches, as all of these participants (except the one who canceled and the two who did not notice the prompt) consented to the fake random prompt. Although some of these participants did give a partially correct rationale for responding to UAC prompts, they failed to apply the rationale when responding to the prompts during the user study. These participants consented to fake prompts because they believed it is the result of their action or did not find any risk associated with consenting to the prompt. In contrast, those participants who exhibited at least a partial knowledge of UAC, had developed a more correct response rationale and demonstrated it by canceling the fake random prompt during the study.

Therefore, understanding a security mechanism can lead users to apply it successfully. We found that there is a strong correlation between "partial understanding of UAC" and "safe response to the fake random prompt" ($p < .001$, Fisher's exact test).

Personal variables and motivations

Our participants' computer expertise impacted their understanding of UAC and their responses to its prompts. Those who did not confirm the fake random prompt had a high (58%) or medium (42%) level of computer expertise. We found that knowledge does play a role, but it still does not guarantee safe actions as 22% of participants with a high level of expertise consented to the fake random prompt.

We also found that understanding a security mechanism impacts users' motivation for applying that mechanism. Most of the PartiallyCorrect participants (79%) preferred to keep the prompts enabled while more than half of the Incorrect participants preferred to disable them.

Some other factors also impacted our participants' motivation for paying attention to UAC prompts. For example, the attitude of Windows 7 participants was impacted by their prior experience in Windows Vista as participants who disabled UAC in Vista followed the same approach in Windows 7. Also, users should not perceive that security mechanisms overlap with each other, otherwise they start to ignore one of them; some participants ignored UAC because they believed their anti-virus software can keep them informed about security risks.

3.4.2 Low-privileged user account

None of our participants used a low-privileged user account on their Windows laptops. This shows that the LUA approach has not been effective in supporting users to follow the PLP.

Communication delivery

The low-privileged user account approach is not communicated to users. The use of LUA is only advised in the online documentation of Microsoft. Our results reveal that 91% of participants were unaware of this principle of computer security,

and all used admin accounts on their systems. A failure in communication left many participants unaware of the benefits of using low-privileged accounts or the risks of high-privileged ones. Our participants' understanding of a low-privileged user account was limited to its restrictions in modifying and managing computer systems, and they perceived using a high-privileged account as a convenient way of working with their computer.

Comprehension and application

Most participants had a partial understanding of the differences between admin and non-admin accounts; 71% mentioned this understanding, and 87% demonstrated it by creating an appropriate user account in the user study task and providing a reasonable rationale for choosing the account type. Even though 42% understood that a low-privileged account is sufficient for most daily tasks, they did not apply this understanding to their own computer usage. Also, while 51% were aware that a high-privileged account allows them to make critical changes to the computer and should be used by trusted users for admin tasks, they could not transfer this knowledge to the potential for malware to compromise their system and perform critical changes on it. This shows the current user account model of Windows operating systems does not convey the security risks associated with using each user account type. A different user account model is required to address this problem.

Personal variables and intentions

Only four participants explicitly demonstrated an understanding of the security risks associated with using admin accounts. Three were participants with a high level of computer expertise and one had a medium level. However, they still preferred using admin accounts because of the ability to modify their systems easily. They were not motivated to consider using a low-privileged user account to avoid such security risks; instead, they relied on their expertise or use of security software.

Not surprisingly, we found that our participants' prior experience with low-privileged user accounts in different contexts appeared to impact their knowledge about these user accounts and their motivation to use them on their personal com-

puters. Although 62% had prior experience with using low-privileged accounts, all but two of these participants (who were also Linux users) were not sufficiently motivated to use such accounts on their Windows PCs.

3.4.3 Principle of least privilege

Prior to the UAC implementation in Windows, users had to use LUA to follow the PLP. However, with the introduction of UAC, if users keep UAC enabled and respond to the prompts correctly, regardless of the type of their user account, they will follow the PLP. If they disable UAC, the type of their user account determines whether they follow or violate the PLP.

Our study shows how well the PLP was followed by our participants. Since all our participants used admin account on their laptop, their use of UAC determines whether they follow the PLP or not:

1. Violated: At least 69% violated the PLP. These are participants who either had disabled UAC and used an admin account (20%) or who consented to the fake random prompt (49%). We chose to use the response rate to the fake random prompt (instead of the fake installation prompt) to determine whether participants respond to UAC correctly in order to be more conservative. A higher bound would be 93% in violation because 73% of all participants (94% of those participants who received both fake installation and real prompt) consented to both the fake and real installation prompt.

2. Followed: Only 27% followed the PLP because they canceled the fake random prompt.

3. Can not be judged: 4% did not notice and respond to fake random prompt.

3.4.4 Recommendations

Based on the findings of our study, we offer the following recommendations to operating system developers for improving the UAC and LUA mechanisms.

Informative content – The UAC prompt should communicate its purpose and information about the program which triggers the UAC to users so that they can

respond to prompts correctly. In our study, those who understood the purpose of prompt could respond more correctly than those who did not have such understanding. The risks and consequences of consenting to a UAC prompt should be conveyed to users so that they can make better decision. The text of the warning plays a significant role in conveying the risk to the user. For example, this mechanism is used in the SSL warnings of Firefox browser.

Selective occurrence – The UAC prompt should include an appropriate level of intelligence to minimize its occurrence. Otherwise, users start to ignore the prompts because of habituation. For example, UAC can remember user's responses to each prompt and avoid triggering the same prompt in the same context. Approaches similar to this are implemented in Smart UAC [30] and Norton UAC [20], however, in these approaches the user has to ask the UAC to stop triggering the prompt in specific contexts.

Integrated solution – Users perceive UAC as an redundant solution because they believe their anti-virus or personal firewall provides the same security functionality. Users may reap a greater benefit from security solutions if their functionalities are integrated so that misconceptions in security coverage do not arise. Providing a package that integrates different security functions might reduce confusion and misconceptions about the protection provided by specific software. This is also in line with the findings of Dourish et al. [6] that "a technology deployed to solve [just] one problem" may not be appropriate for end-users.

Risk communication – The risks of high-privileged user accounts and the benefits of low-privileged ones should be conveyed to users; otherwise, users will not be motivated to follow the principle of least privilege by using low-privileged accounts for daily tasks. Currently, users perceive admin accounts as a convenient way for applying changes on their computers. Their mental model of different user account types is limited to the rights and capabilities of each account type. Operating system developers should improve the users' mental model by conveying the security issues of account types to users.

Convenient usage – Using low-privileged user accounts should be convenient for users as they perform legitimate and informed actions on their PCs. Otherwise, they may quit using low-privileged accounts after facing restrictions.

Default settings – Although Microsoft advises users to create a non-admin

account, the initial and only account created during OS installation is an admin account. In addition to this account, a low-privileged account should be created; users should be encouraged to use it for daily work.

3.5 Limitations

The goal of our research was to study the users' understanding, behavior, and challenges in applying the PLP, which targets every end-user of computer systems. Due to the challenges of conducting studies that investigate the users' normal behavior, there are some threats to validity of our results.

First, it was difficult to study a participant sample that represents the real user population. In our study, the participant recruitment was more challenging than usual. The middle-age people tended to use older laptops, which often had previous versions of the Windows operating system. Recruiting participants for Windows 7 was particularly difficult because it was not yet widespread in the general community; most respondents were computer science or engineering students who had upgraded their system to this operating system. As a result, 60% of our participants were recruited from students. Therefore, there are some threats to external validity of our study as our sample does not fully represent the real user population. However, compared to similar studies in usable security community [36] in which all participants were recruited from university students, our sample had a higher diversity.

Also, some respondents to our recruitment notice were concerned about installing applications on their laptops. Therefore, there are some threats to internal validity of our study as some users who were highly cautious about the security of their systems did not take part in the study.

Conducting the study in the lab environment increased the threat to the external validity of results since participants were not in their natural conditions of working with the computer and were observed by a researcher. Also, their screen and voice were digitally recorded. To decrease these threats, we conducted the study on participants' laptops and did not give them detailed instructions for performing the tasks. Instead, we asked them to do the tasks as they usually do in their normal computer usage. Also, they were informed that the task completion is not the

goal of the study and they can quit or not perform any of the study tasks. There is evidence that our decisions were effective as some participants refused to perform some tasks because they had concerns such as decreasing the performance of their system, inconsistency of installed application with their current applications, and no familiarity with the applications that were asked to be installed. We also decreased the social threats to construct validity of the study by not revealing the purpose of the study to participants so that their behavior was not changed based on the study purpose.

We had to rely on participants' self reports during various points of the study, which sometimes led to incomplete or inaccurate data. For example, our understanding about the challenges users face in using LUA and UAC approaches is not complete because participants could not accurately remember their previous experiences outside of the study environment. Also, our assessment of participants' computer knowledge and expertise was mostly based on self report, but some participants may not have had a correct understanding of their expertise or may not have reported it accurately.

Since the prior experience of users in using LUA and UAC approaches could have influenced their behavior in the study, we probed their prior experience in using these approaches and contrasted it with their behavior in the study.

We did not study people at their workplaces; because, in addition to the difficulty of recruiting people at their workplaces, their workplace user accounts are usually low-privileged due to the computer security policies of organizations. We are mostly interested in users' behavior when they are not forced to follow any imposed policy.

3.6 Conclusion

Our user study and interviews with 30 Windows Vista and 15 Windows 7 participants provided a rich description of the practices of users in applying the principle of least privilege. Our analysis revealed the reasons why this principle is often not followed by users. We studied users' motives, understanding, behavior, and the challenges they face when they use two implementations of this principle: UAC and low-privileged user account. We found that 69% of our participants did not

employ the UAC approach correctly as they either disabled it or consented to any UAC prompt that arose when they were in the context of doing an action, especially when they initiated an action themselves. Most participants had an incorrect understanding of UAC and responded to prompts incorrectly.

All our participants used an admin account on their laptop. Although 71% had a partial understanding of the limitations and rights of each user account type, 91% of participants were not aware of the security risks of high-privileged accounts or the security benefits of low-privileged ones. Also, while 62% had experienced a low-privileged user account, they were not motivated to use it on their own laptops because of the limitations they had faced using these accounts.

Based on our results, we recommend conveying the purpose and benefits of LUA and UAC to users, raising UAC prompts in fewer situations, integrating UAC functionality with other security software, providing appropriate information about the program triggering the UAC in the prompt, providing users with default low-privileged accounts, and making the use of low-privileged account convenient in order to ensure that users continue to use them.

Chapter 4

Information Content for Assessing the Risk in Privilege Elevation

4.1 Introduction

The purpose of UAC prompt is to make the users aware that a program is trying to make administrative (or admin for short) changes on their computer and give the users the opportunity to stop the program. Because most types of malware need admin privileges for their execution (e.g. changing the system configuration, installing a program, writing in the system directories), the UAC prompt can be a layer of defense against them. In a perfect world, the operating system would stop any malicious program that intends to apply admin changes. However, there are many threats that the operating system cannot detect with 100% accuracy and false positives may exist. One solution is to warn the user about possible security threats. It should be noted that the UAC prompt or similar privilege elevation prompts in other operating systems do not provide a security boundary or direct protection, but gives the user a chance to verify the admin changes before allowing them to take place.

One interesting finding of our first study was that when users understand the

purpose of the UAC prompt and the risks it aims to mitigate, they can respond to UAC prompts correctly. They also appreciate the security protection provided by the UAC prompt and prefer to receive such prompts. If users respond to UAC prompts correctly, the PLP can be followed without sacrificing the convenience of working with admin accounts.

Due to the important role of UAC prompts in enforcing the PLP and users' ability to respond correctly to these prompts (if they understand the prompt purpose and the conveyed risk), we decided to improve the UAC prompt by applying one of our recommendations in the first study; providing an informative content on the prompt.

Prior research [33] showed that there are five main factors that must be considered when the effects of warnings are examined.

1. The information content of the warning
2. The format in which the information content is presented
3. The purpose of the warning
4. The criteria by which the success or failure in achieving the warning's purpose is to be judged
5. The characteristics of the audience of the warning

However, published research on the effects of warnings has focused more on the form, format, and structure of warnings than on the information content. Research that focuses both on the content and presentation format of warnings cannot distinguish the effect of the presentation from the effect of the information content [33]. Therefore, in this part of our research, we focus on the information content of the UAC prompt. The goal of our research is to determine the information items that can assist users in assessing the risk of privilege elevation more accurately so that they can respond to UAC prompts correctly. We also aim to identify the information items that are beneficial for users with different levels of computer knowledge and expertise.

Our first study showed that the poor response of participants to UAC prompts is correlated with one or more of the following factors:

1. Users are not sure whether the UAC prompt is the result of their action or not. When users are in the context of doing an action, they consent to the UAC prompt, because they think the prompt is the result of their current action.

2. Users are not aware of the purpose of the prompt and the risk the UAC prompt intends to mitigate.

3. Users are habituated to UAC prompts and do not pay close attention to them.

In this research, we aim to address the first two problems by providing an informative content in the prompt that assists users in understanding the purpose of the prompt and responding to it correctly. To this end, we included thirteen different information items on the UAC prompt and investigated how users understand and utilize these information items for responding to the UAC prompts in different contexts. We conducted a user study with 48 participants who had diverse demographics characteristics. We asked our participants to respond to eight different UAC prompts and explain how they used the information items of the prompts for responding to them. Four of the prompts simulated a benign scenario and the other four simulated a malicious scenario. All the prompts included all thirteen information items but the content of items differed in each prompt. Our results showed most participants understood the purpose of modified UAC prompt correctly. The items that were understood, used correctly and preferred by most participants included program name, origin, description, digital certification, changes that the program applies and the result of program scan by anti-virus. Our participants with a high level of computer knowledge and expertise also found the program extension and bundled program as useful information items. Some of the other items were also understood by many participants but these items will be more beneficial for users if they are presented in specific contexts. Therefore, we recommend changing the UAC prompt to a context-based prompt that presents a subset of information items to users in each context. The guidelines for selecting this subset in each context are presented in this chapter.

In the remainder of this chapter, we first describe our threat model followed by describing the information content we considered for the UAC prompt. Section 4.4 and section 4.5 present the methodology of our study and its results respectively.

Section 4.6 provides a discussion of our results, as well as recommendations for designing the UAC prompt. Section 4.8 is the conclusion of this chapter.

4.2 Threat model

In our threat model, the adversary goal is to gain unauthorized admin access to the user's computer. For this purpose, the adversary runs or tricks the user to run a malicious executable file on the user's computer. This executable file runs as an independent process and performs an operation that needs admin privileges. The execution of this operation triggers a UAC prompt. The malicious executable file can be transferred to the user's computer via different means such as:

1. Downloads that happen without the user's knowledge. For example, by visiting a website that is infected by a malware which drops malicious files on the visitors' computers.

2. Downloads authorized by the user without understanding the consequences. For example, downloading audio, video, etc. files from peer to peer sharing websites that embed malicious executables.

3. Opening an infected email attachment. For example, opening pdf, excel, word files that include malicious macros.

4. Installing bundled program. For example, user installs a benign program, but other unwanted programs are also installed and executed on the computer.

The report by SANS shows that email attacks and compromised web sites are two primary methods of infection for compromising the computers that have Internet access. The report shows that 60% of total attacks on the Internet target the web applications. Such attacks transform the web applications to web sites that deliver malicious content.

We do not consider the following attacks in our threat model:

1. If a malicious code is introduced (or injected) into a running benign process and the malicious code requires the same admin privileges as the benign process, the malicious code can use the privileges of the benign process to

execute its actions. In this scenario, no UAC prompt is triggered for the malicious process.

2. If malware is bundled with a benign program and it triggers UAC prompt before the benign program, user may consent to this UAC prompt assuming that the prompt is triggered by the benign program. However, if the malicious program triggers the UAC prompt after the benign program, user can conclude the second prompt is not related to the benign program installation since the benign prompt indicates [1] that no other program is about to run after it.

3. If malware runs without performing an action that requires admin privileges (e.g. sending email from the user's account), no UAC prompt will be triggered.

4.3 Information content of the UAC prompt

This section introduces the assumptions we made for choosing the information content of the UAC prompt, our strategy for selecting the content and the information items considered for the prompt.

4.3.1 Assumptions

We made the following assumptions for selecting the information content of the UAC prompt:

1. The operating system can detect how the program that has triggered the UAC prompt is launched. This can be achieved by back tracking the process tree of each program. Such data exists in the operating system, however, the operating system developers need to collect the data and process it. The launcher of the program can be one of the following factors and is identified by the operating system.

[1] One of the considered information items indicates whether any other program will execute after the current program that has triggered the UAC prompt

(a) User: The user can launch a program via different means such as double clicking the program file.

(b) Another program: The program triggering the UAC prompt may be launched by another program.

(c) Unknown: The operating system may not be able to determine how the program is initiated.

Our previous study showed that users have difficulty in responding to UAC prompts because they are not sure whether the prompt is the result of their action or the prompt is triggered via other means. Therefore, the information about how the prompt has been triggered (e.g. by the user action or another program) can be useful for users.

However, this information is not sufficient for responding to the UAC prompt. As an example, a prompt that is triggered for "Automatic Java Update" is not initiated by the user but the update is a valid task to perform. On the other hand, the user may initiate a program that is malicious.

2. Program developers are required to specify the required privileges needed for running their program in the program manifest. The manifest is an XML file that explains the shared and private side-by-side assemblies that an application should bind to at the run time [1]. Such approach is used in Android applications. If a program tries to do an action that requires admin privileges and is not listed in the program manifest, the program execution will be stopped by the operating system. It is an additional effort for developers; however, if the operating system intercepts privilege elevation requests from processes and triggers the UAC prompts for each request, the user will receive multiple prompts for performing a task that requests privilege elevation several times.

3. Program developers are required to list the programs that will be installed or executed after their program in the program manifest. This information item will be used to inform the user about the programs that will be installed or executed after the program triggering the UAC prompt.

4. The UAC prompt is triggered whenever a program that requests admin privileges in its manifest, is about to execute. In our study, the set of tasks that trigger a UAC prompt is the same as tasks that trigger a UAC prompt in Windows Vista. Therefore, this assumption is already implemented.

4.3.2 Strategy for proposing the content of the prompt

Our previous study revealed that the following information is missing from the current UAC prompts:

1. Information about the factor that triggers the UAC prompt: Users do not recognize how the UAC prompt is triggered. They are not confident whether the prompt is triggered as a result of their action or by the other means. When a UAC prompt is raised, users are usually in the context of running a program. Therefore, they may assume the prompt is related to the program they are running. Consequently, it is beneficial for the users to indicate how the UAC prompt is triggered.

2. Information about the purpose of the UAC prompt: Users do not know the purpose of the UAC prompt and the risk it aims to mitigate. Therefore, it is necessary to show users information about the risks of running an application.

To address these shortcomings, we included several information items on the UAC prompt to inform users about the risks of consenting to the UAC prompt and the factor triggering the prompt. Also, the question which is asked of users on the prompt aims to inform the user about the purpose of the prompt.

4.3.3 Prompt information content

We changed the question on the UAC prompt to the following question:

The below program wants to apply important changes on your computer. What would you like to do?

The response options are “Continue program” and “Stop program”. This question shows why a UAC prompt is triggered on the computer. The question of

Windows 7 UAC prompt is similar to the question that we used in our study, but it uses a different wording. The question in the MAC OS X and Ubuntu focuses more on the authentication. Also, the UAC prompt in Windows Vista asks for user permission to continue the program and does not inform the user about the changes. During the study, we investigated how well participants understood the question on the prompt.

The following is the list of the prompt information items. Some of these items already exist in the operating system; however, some items are new and should be collected by the operating system. For each item, we describe how this information item can be helpful for user in responding to UAC prompt. Each item has a label which we use to refer to the item in the thesis. The label is mentioned in the parentheses beside the item name.

1. **Basic information about the program**

   (a) Program name (Name): The name of the program that triggered the UAC prompt, is the program identification and is inevitable to present to users. For example, the program name can be AdobeReader or GameSetup.

   (b) Program extension (Extension): The extension of the program shows the type of the file that is about to apply changes on the user's computer. The extension can be .exe, .msi, dll, etc. In an attack scenario, an attacker may hide the extension of the file and trick the user to open the file (e.g. an email attachment which is an executable but is represented as a Microsoft Word document). If the user observes the real extension in the prompt, she may notice such discrepancy.

   (c) Program path (Path): The program path shows the location of the program on the computer. If a program is not initiated by the user, the path can help the user to identify the location of the program. An example path can be: "C:\Users\John \Downloads".

   (d) Creation date/time (Time): This item shows the date and time that the program is created on the user's computer. If the creation date/time of the program does not match the user's action, the prompt might

have been triggered by a malicious program. For example, if the UAC prompt shows the program creation date as 20 seconds ago and the user has not initiated any action in this period which results to the file creation, the program may be a malware.

(e) Program origin (Origin): This item specifies the source of the program. For example, the origin can be the URL of a website where the program is downloaded from or an external device. Users may find some origins more trustworthy than the others. Therefore, this item can impact their decision process.

2. **Program actions**

(a) Program description from a trusted source (Description): This item provides the user with a short description about the program functionality and features. This description is retrieved from a source trusted by the operating system. For instance, the operating system can require application developers to submit their application for review to the operating system related website. If the application is benign, the website posts a description for the application. If no such description exists, the user is notified. For example, a short description about an Adobe update could be: "This program adds one or more new features to your AdobeReader program."

(b) Changes to apply (Changes): This item shows the changes that the program applies on the computer if it is executed. It shows the files or settings that will be changed on the computer. Some explanation is added to these changes so that average users can understand it. As an example, when a program is about to install on the computer, the following changes can be displayed on the prompt: "To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system."

(c) Previously applied changes (Pre. changes): This item shows the changes that the program has already applied on the computer if it was previously executed on the computer. This item can help users to recall the

program if it was executed on their computer previously. This item has a content similar to the "Changes" item.

3. **Program launch**

   (a) Program executed by (Executor): This item shows how the program is launched (by the user, by another program, unknown). Such information can help user to identify what factor has triggered the prompt.

   (b) Number of previous executions (Pre. executions): This item shows the number of times the program has been executed on the user's computer before as well as the previous user's responses. Such information can help the user recall the program if it was previously executed or consider the program more carefully if it is the first time that the program is executed on the computer. For example, a program might have been executed twenty times and the user allowed its execution all the twenty times.

   (c) Trusted programs to be executed after this program (Bundle): This information item shows whether any other program is bundled with this program and will be executed after it. If a malicious program is bundled with a benign program and the malicious program triggers a prompt after the benign program, since the benign program prompt has indicated that no other program will run after it, the user can conclude that the second prompt is not related to the benign program execution. The content of this item will be "None" if no program is bundled with the program triggering the UAC prompt; otherwise it shows the name of the program that will be executed after the current program.

4. **Program security features**

   (a) Program digital certification (Certificate): This item shows whether the program has a valid digital certification issued by a CA trusted by Windows. It also shows the program publisher that the certificate is issued to. For example, the prompt for Adobe update shows the following for this item: "It has a valid certification issued to Adobe Systems, Inc." If

a program has a valid certification, users can conclude that the program is not changed by an unauthorized party.

(b) Result of scan by an anti-virus (Virus): The program is scanned by the anti-virus of the user's computer (or an online anti-virus) and the scan result will be presented to the user. If the anti-virus finds any malware in the program, the user should not run the program. However, a program which is reported as benign by anti-virus may still be malicious. (Zero-day attack)

5. **Other users' decisions**

   Information about how other users responded to the same prompt can help the user in responding to it. However, we do not include such information on the prompt because the effect of including such information is already studied [2][11]. It has been shown that presenting this type of information impacts the user's decision. However, it is not guaranteed that following the decision of other users is the correct action to take. Therefore, the challenge is to distinguish the decision of expert users from other users' decisions and present it to the users.

4.4 Methodology

Our research goal in this study was to identify a set of information items to include in a UAC prompt so that participants respond to the prompt correctly. We also aimed to identify which information items are more beneficial for participants with a specific level of computer knowledge and expertise.

Specifically, our research questions in this study were:

1. Do our participants understand the information items on the UAC prompt?
2. Which information items do participants utilize to respond to UAC prompts in different contexts?
3. What other information items are required to be included on the prompt from participant's perspective?

4. What is the correlation between participants' information utilization and participants' level of computer knowledge?

To answer these questions, we investigated how participants understand and utilize the information items on the prompt for responding to it in different contexts. Based on our results, we identified which information items were or were not beneficial for participants with a specific level of computer knowledge.

To examine participants' understanding of information items on the prompt, we asked them open-ended questions. Asking open-ended questions and multiple choice questions are two common approaches for examining users' understanding of warnings [14, 53]. Since open-ended questions provide more information about users' conceptions and misconceptions [49], we used this approach in our study.

To evaluate participants' behavior in responding to prompts, the ideal solution is conducting a naturalistic longitudinal study [53]. However, since UAC prompts are triggered infrequently, this approach will be very time and labor consuming [8]. The other approach is conducting a laboratory study. This approach also has some limitations as it is challenging to simulate a believable risk situation for participants [31] and participants are not usually motivated to treat the risk situation in the study the same as a real risk situation [8].

Since our focus in this study was to examine how participants understand and utilize the information items of the prompt, we designed our study not to examine their realistic behavior, but rather to examine how the information items on our prompt can assist participants in deciding about privilege elevation. For this purpose, we asked participants to respond to a set of UAC prompts and indicate how they used different information items of the prompts for responding to them. Although this approach relies on self reported data, it can provide us with participants' understandings and misconceptions about different information items of our prompt. It also indicates whether participants were able to utilize these items correctly or not. Since it is advised in warning science literature to provide context for participants when examining their understanding and behavior in responding to warnings [53], we provided a scenario for each of the prompts that participants would respond in the study. We used the role-playing approach and asked participants to read the scenarios and assume they are in such a situation on their own

Type	Prompt label	Description
Benign	B_Bundle	Install a bundled program
	B_AdobeUpdate	Install an auto update from Adobe
	B_NotCertified	Install a program with description but no certification
	B_PreUsage	Run a previously used program needing access to firewall
Malicious	M_ExeAttachment	Open an executable attachment that writes files in the System dir. and connect to a website.
	M_DriveBy Download	Drive-by download when visiting a website, writing files in System dir.
	M_Virus	Install a program from USB infected with virus
	M_Bundle	Install a malicious program bundled with a benign one

Table 4.1: Scenarios description

computer.

4.4.1 Study design

In the study we asked participants to respond to eight different UAC prompts. We developed a scenario for each prompt to provide context for participants. Four scenarios simulated an attack scenario and four others were benign privilege elevation scenarios. Table 4.1 shows a brief description about each scenario and the label of the corresponding prompt. We use these labels to refer to the prompts in the thesis. These scenarios happen usually in the daily computer usage.

All these prompts included all thirteen information items that we considered for the UAC prompt, however their content changed in each prompt. Each prompt was aimed to evaluate whether participants can utilize some particular information items on the prompt. Table 4.2 shows which items were expected to be used in each prompt. As this table shows, all items were expected to be used when considering all the eight prompts. Some items were expected to be used in more than one prompt. Tables 4.3, 4.4 and 4.5 show the content of each item in all eight prompts.

Since all participants responded to all prompts, our study was a within-subject study. We exposed participants to all prompts to investigate how different participants respond to each prompt. To reduce learning effects, fatigue and presentation sequence effects introduced by a within-subject design, we counterbalanced the

presentation order of prompts. The best approach for counterbalancing numerous conditions is “balanced Latin square”. For eight prompts, the balanced Latin square is shown in table 4.6. Each row represents an order. It can be seen that each prompt appears precisely once in each row and column. Furthermore, each prompt appears before and after each of the other prompts an equal number of times. For example, prompt B_AdobeUpdate follows prompt B_Bundle two times and it also precedes prompt B_Bundle two times.

	Benign Prompts				Malicious Prompts			
Information Item	B_Bundle	B_Adobe Update	B_Not Certified	B_PreUsage	M_ExeAttachment	M_DriveBy Download	M_Virus	M_Bundle
Name	✓	✓					✓	✓
Extension					✓	✓		
Path	✓	✓	✓			✓		
Origin		✓	✓			✓		✓
Time	✓	✓	✓			✓		
Description		✓	✓		✓	✓	✓	✓
Changes	✓	✓	✓		✓	✓		
Pre. changes				✓				
Executor	✓	✓	✓	✓		✓		
Pre. executions				✓				
Bundle	✓							✓
Certificate		✓						✓
Virus	✓	✓	✓	✓			✓	

Table 4.2: Information items that were expected to be used by participants in each prompt

Prompt	Name	Extension	Path	Origin	Time
B_Bundle_1	Listen Music	.exe	C:\Users\John \Downloads	http://www.music.players.com	1 min. ago
B_Bundle_2	Audio Codes	.exe	C:\Users\John \Downloads	http://www.musicplayers.com	2 min. ago
B_Adobe Update	Automatic AdobeReader Update	.exe	C:\Program Files \Adobe	http://www.adobe.com	Last week
B_Not Certified	Chat	.msi	C:\Users\John \Downloads	http://www.download.cnet.com	1 min. ago
B_Pre Usage	Game	.exe	C:\Temp	http://www.gameforyou.com	1 year ago
M_Exe Attachment	Scenery Photo	.exe	C:\Users\John \Downloads	http://www.gmail.com	30 sec. ago
M_DriveBy Download	Windows Media Player	.msi	C:\Users\John \AppData \Local \temp	http://www.funsongs.com	2 min. ago
M_Virus	Text Editor Setup	.exe	E:\Program	USB flash drive	NA
M_Bundle_1	Chess	.exe	C:\Users\John \Downloads	http://www.fungame.com	15 sec. ago
M_Bundle_2	Setup	.exe	C:\Users\John \Downloads	http://www.fungame.com	20 sec. ago

Table 4.3: Content of information items in each prompt - part1

Prompt	Description	Executor	Pre. Executions	Bundle	Certificate	Virus
B_Bundle_1	A program to play audio or video file.	You	Zero	Audio Codes .exe	A valid certification issued to ListenMusic, Inc.	No virus
B_Bundle_2	No description is found.	Listen Music .exe	Zero	None	No certification.	No virus
B_Adobe Update	This program adds one or more new features to your AdobeReader program.	Windows Start up Program	Zero	None	A valid certification issued to Adobe Systems, Inc.	No virus
B_Not Certified	A program for chatting with your friends.	You	Zero	None	No certification	No virus
B_Pre Usage	No description is found.	You	Continue 98 times	None	No certification	No virus
M_Exe Attachment	No description is found.	You	Zero	None	A valid certification issued to Photo Systems, Inc.	No virus
M_DriveBy Download	No description is found.	Not known	Zero	None	A valid certification issued to FunSongs, Inc.	No virus
M_Virus	No description is found.	You	Zero	None	A valid certification issued to Text Edition, Inc.	Virus found
M_Bundle_1	A program to play chess against computer or a friend.	You	Zero	None	A valid certification issued to ChessRally, Inc.	No virus
M_Bundle_2	No description is found.	You	Zero	None	No certification	No virus

Table 4.4: Content of information items in each prompt - part2

Prompt	Changes	Pre. changes
B_Bundle_1	To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system	None
B_Bundle	To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system	None
B_Adobe Update	To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system	None
B_Not Certified	To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system	None
B_PreUsage	Adding itself temporarily to unblocked programs of your firewall	Same as changes
M_Exe Attachment	Changing system functionality by writing files in C:\Windows\System32, started automatically and sending or receiving information to www.photoviewer.com	None
M_DriveBy Download	Changing system functionality by writing files in C:\Windows\System	None
M_Virus	To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system	None
M_Bundle_1	To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system	None
M_Bundle_2	To be added to system programs by writing its files in C:\Program Files and storing its configuration settings in the system	None

Table 4.5: Content of information items in each prompt - part 3

Order	Scenarios							
1	B_Bundle	B_Adobe Update	M_Bundle	B_Not Certified	M_Virus	B_PreUsage	M_DriveBy Download	M_Exe Attachment
2	B_Adobe Update	B_Not Certified	B_Bundle	B_PreUsage	M_Bundle	M_Exe Attachment	M_Virus	M_DriveBy Download
3	B_Not Certified	B_PreUsage	B_Adobe Update	M_Exe Attachment	B_Bundle	M_DriveBy Download	M_Bundle	M_Virus
4	B_PreUsage	M_Exe Attachment	B_Not Certified	M_DriveBy Download	B_Adobe Update	M_Virus	B_Bundle	M_Bundle
5	M_Exe Attachment	M_DriveBy Download	B_PreUsage	M_Virus	B_Not Certified	M_Bundle	B_Adobe Update	B_Bundle
6	M_DriveBy Download	M_Virus	M_Exe Attachment	M_Bundle	B_PreUsage	B_Bundle	B_Not Certified	B_Adobe Update
7	M_Virus	M_Bundle	M_DriveBy Download	B_Bundle	M_Exe Attachment	B_Adobe Update	B_PreUsage	B_Not Certified
8	M_Bundle	B_Bundle	M_Virus	B_Adobe Update	M_DriveBy Download	B_Not Certified	M_Exe Attachment	B_PreUsage

Table 4.6: Presentation order of prompts

4.4.2 User study protocol

The study procedure was as follows:

1. Brief the participant about the study: The participant was informed that we have developed a new computer prompt and she is supposed to read the information on the prompt and respond to the prompt. In order to motivate participants to read the prompt and answer correctly, we told them that three participants with the best performance would be given a prize. The orientation script is shown in table 4.7.

2. Getting the consent: The consent form outlined the purpose of the study, length of the study, study procedure, the participant's right to withdraw from the study without consequence, and provided an assurance of confidentiality and anonymity of personal data.

3. Performing tasks: Participants read eight scenarios one by one and then observed and responded to each prompt, which is triggered in such a scenario. A sample of one task and its corresponding prompt is shown in Appendix B.2. The prompts were displayed on the researcher's computer. All prompts had the same information items however, the content of information items changed in each prompt. The participants were informed in the beginning of the study that they can skip a scenario if they do not understand it.

4. Post task questionnaire: After responding to each prompt, participants filled a questionnaire to answer the following questions. The questionare, which is shown in Appendix B.2, included the following questions:

   (a) Usefulness rating: Participants indicated to what extent each information item was useful for them in responding to the prompt using a Likert scale from 5 (very useful) to 1 (not useful). Also, they could indicate if they did not pay attention to an item or did not understand it.

   (b) Perceived risk: Participants indicated the level of hazard they perceived when they read the information on the prompt using a Likert scale from 5 (very risky) to 1 (not risky).

(c) Impact rating: Participants indicated the impact level of each information item on their decision. For this purpose, they showed whether each item encouraged them to continue or stop the program. They also indicated how strong this impact was using a Likert scale from 5 (very strong impact) to 0 (no impact).

Although it has been shown that ranking is preferable to rating when measuring values as rankings are less sensitive to user-response variations, we choose rating in our study to measure the usefulness and impact of information items. We did so because ranking thirteen items sixteen times was a challenging task for participants. Also, we could not use ranking for measuring the perceived risk of prompts as it was unlikely that participants remember all the eight prompts at the end of the study.

5. Exit interview: After responding to all prompts, participants participated in an interview to describe their understanding about the prompt and its information items, their opinion about the most useful items and their requirement for other information pieces. The interview questions are shown in Appendix B.3.

4.4.3 Task scenarios

In this section, each scenario of the user study and its corresponding prompt is described. We also explain the rationale for including each scenario and the information items that participants were expected to utilize in each prompt. The label of prompt corresponding to each scenario in mentioned is the title of the scenario.

Benign scenarios

Scenario 1 (B_Bundle):

This scenario simulates two benign program installations. The second program is bundled into the first one. Therefore, two prompts are triggered consecutively. The field of "Bundle" in the first prompt mentions the name of the program of the second prompt to indicate a second prompt follows. Also, the "Executor" field in the second prompt had the name of the program in the first prompt.

Orientation Script
I am studying how users understand and utilize a set of information items on a computer message. You see a sample of this message on the screen now. I ask you to read a set of scenarios. After reading each scenario, you see one of these messages on the screen which is triggered in such a scenario. Please assume you are in such a scenario on your own computer and have received such a message. Please read the information of the message carefully and choose one of the response options (continue or stop program). Also, it would be helpful for me if you can think aloud while you are making decision. Then you fill a questionnaire about the message. If you do not understand a scenario, please let me know. At the end of the session, I ask your feedback about this kind of computer message. We give a prize to three participants who provide the correct answers in the study. During the interview, I record your voice to capture the details that I might miss during my note taking. Do you have any questions? If not, then let's begin. To begin, first, please first read the consent form and sign it.

Table 4.7: Orientation script

In this scenario, we investigated how participants respond to the prompt corresponding to the bundled program installation and whether they can use the information item "Bundle" in the first prompt for responding to the second prompt.

The scenario description was:

You want to listen to music but your music player program does not work. So, you decide to install a different music player. You visit musicplayers.com and download the ListenMusic program from the website. When you double click the installation file, you see the prompt shown on the screen now. Please read the prompt carefully and respond to it.

In addition to the "Bundle" field in the first prompt, the following items in the second prompt can assist participants in concluding that a benign program is about to apply changes on their computer:

1. Name: The name of the program was mentioned in the first prompt.
2. Path: The program was executed from an expected location.
3. Time: The program was created on an expected time.

4. Changes: The changes are normal for a program to be installed.

5. Executor: The program was executed by the program in the first prompt.

6. Virus: Program has no virus.

Scenario 2 (B_AdobeUpdate):

This scenario simulates a benign auto update for AdobeReader. Although user has not initiated this update, it is a benign program to run if the user wants the update.

The scenario description was:

You have downloaded and installed AdobeReader program on your computer last week from www.adobe.com to read PDF files. Today after you turn on your computer, you see the prompt which is shown on the screen now. Please read the prompt carefully and respond to it.

The following information items can assist participants in concluding that a benign program is about to apply changes on their computer:

1. Name: The name of the program is well-known.

2. Path: The participant can verify that the program was initiated from the same directory as AdobeReader was installed.

3. Origin: The program was downloaded from a well-known website.

4. Time: The program was created last week when the user has installed the Adobe Reader program.

5. Description: The program has a description retrieved from a trusted source.

6. Changes: The changes are normal for an update to be installed.

7. Executor: The program is executed by the operating system.

8. Certificate: Program has a valid digital certification issued to ”Adobe Systems”.

9. Virus: Program has no virus.

Scenario 3 (B_NotCertified):

This scenario simulates installation of a program that although does not have a digital certification, it is downloaded from a well-known website and it has a description retrieved from a trusted source. Such scenario happens frequently for users as they install software that does not have a digital certification.

The scenario description was:

You want to chat with your friend. You visit download.cnet.com and download a chat program which it seems has interesting features. When you double click the installation file, you see the prompt shown on the screen now. Please read the prompt carefully and respond to it.

The following information items can assist participants in concluding that a benign program is about to apply changes on their computer:

1. Path: The program was executed from an expected location.
2. Time: The program was created on an expected time.
3. Origin: The program was downloaded from a well-known website.
4. Description: The program has a description retrieved from a trusted source.
5. Changes: The changes are normal for a program to be installed.
6. Executor: The program is executed by the user not an unknown source.
7. Virus: Program has no virus.

Scenario 4 (B_PreUsage):

This scenario simulates the execution of a program that needs access to the firewall. Although the program does not have a digital certification and a description from a trusted source, the user has previously used it.

The scenario description was:

You are testing the programs of your computer. You run a program and see the prompt shown on the screen now. Please read the prompt carefully and respond to it.

The following information items can assist participants in concluding that a benign program is about to apply changes on their computer:

1. Pre. changes: The changes that the program is about to apply on the computer, is the same as the previously applied changes. So the participant can conclude such changes have been applied on the computer before.

2. Executor: The program was executed by the user not an unknown source.

3. Pre. executions: Since program has been executed before, the user can remember the program more easily.

4. Virus: Program has no virus.

Malicious scenarios

Scenario 5 (M_ExeAttachment):

This scenario simulates an attack scenario where the attacker tricks the user to open a malicious attachment. Opening the attachment leads to dropping and execution of an executable file which applies admin changes to modify the functionality of the system. We assume the attacker has utilized a zero-day vulnerability so the anti-virus cannot detect the malware and has stolen a valid digital certification to sign the malware. An attack similar to this scenario has occurred recently [18].

The scenario description was:

You have received an email in your Gmail account with an attachment from one of your friends. The attachment is a photo from scenery. You open the attachment and observe the prompt which is shown on the screen now. Please read the prompt carefully and respond to it.

The following information items can assist participants in concluding that a malicious program is about to apply changes on their computer:

1. Extension: The extension is .exe which shows opening a photo had caused the execution of a program.

2. Description: Since there is no description available for the program, it might be risky to execute the program.

3. Changes: The participant is expected to understand that opening a photo should not lead to changing the functionality of her computer.

Scenario 6 (M_DriveByDownload):

This scenario simulates a drive-by download attack in which the attacker has introduced a malicious code into a website. When users visit the website, an executable file is dropped and executed on their computers. We assume the attacker has stolen a valid certification to sign the malware and has utilized a zero-day vulnerability so that the anti-virus cannot detect the malware.

The scenario description was:

You are visiting the SongLyrics.com to download a song. You download a song and while you are waiting for the download to finish, you see the prompt which is shown on the screen now. Please read the prompt carefully and respond to it.

The following information items can assist participants in concluding that a malicious program is about to apply changes on their computer:

1. Extension: The extension is .msi which shows visiting a website has caused the execution of a program.

2. Origin: It shows the program has been downloaded from a website that the user is currently visiting.

3. Path: The participant can verify the program was initiated from an unfamiliar directory that she does not use for her downloads.

4. Time: This shows the file has just been created on the computer. Therefore, the file is related to the current actions of user. However, the user has not downloaded or copied any program on the computer.

5. Description: Since there is no description available for the program, it might be risky to execute the program.

6. Changes: The participants were expected to understand that visiting a website should not lead to changing the system functionality.

7. Executor: Since it is not known how the program has been executed, the participant can conclude a program is about to run on her system without her intention.

Scenario 7 (M_Virus):

In this scenario, the program which is about to run is a malicious program and the anti-virus has detected a virus in the program.

The scenario description was:

You received a USB disk from your friend. He has mentioned there is a Text Editor program on the USB which has interesting features. When you start to install this program, you see the following prompt on your screen. Please read the prompt carefully and respond to it.

The following information items can assist participants in concluding that a malicious program is about to apply changes on their computer:

1. Name: The name of the program is not well-known.
2. Description: Since there is no description available for the program, it might be risky to execute the program.
3. Virus: The anti-virus has detected a virus in the program and participants were expected to notice this information item.

Scenario 8 (M_Bundle):

This scenario simulates a bundling attack in which a malicious program is bundled with a benign program. We assume both programs start to run when the user runs the benign program. Also, we assume that the privileges requested by the malicious program are different from the privileges requested by the benign program. Therefore, both programs trigger a UAC prompt. The prompt of the malicious program is triggered after the benign program prompt. Since the "Bundle" field of the benign program prompt shows that no other program is about to run after this program, the participant can conclude that the second prompt is not related to the program installation that she has initiated.

The scenario description was:

You have downloaded a chess game from a website. You start to install the program and see the following prompt on the screen. Please read the prompt carefully and respond to it.

In addition to the “Bundle” field in the first prompt, the following information items can assist participants in concluding that a malicious program is about to apply changes on their computer:

1. Name: The name of the program is not well-known.
2. Origin: It shows that the program is downloaded from a website which is not well-known.
3. Description: Since there is no description available for the program, it might be risky to execute the program.
4. Certificate: Program has no digital certification.

Order		O1	O2	O3	O4	O5	O6	O7	O8	Total
Group Size (N)		6	6	6	6	6	6	6	6	48
Age	Mean	25.17	28.17	38.33	25.17	27	31.83	27.5	35.5	29.5
	Range	19-30	19-39	23-50	18-37	19-41	19-67	19-43	24-60	18-67
Gender	Female	3	3	3	3	3	3	3	3	24
	Male	3	3	3	3	3	3	3	3	24
Student	Yes	4	2	2	3	3	4	3	3	24
	No	2	4	4	3	3	2	3	3	24
Educational Level	Highschool	2	2	2	2	3	3	4	3	21
	Bachelor	3	3	4	4	2	1	1	3	21
	Master	1	1	0	0	1	1	1	0	5
	PhD	0	0	0	0	0	1	0	0	1
Background	Computer	1	1	0	0	0	0	1	0	3
	Science	3	1	2	0	0	1	1	0	8
	Engineer	1	0	1	1	2	2	1	3	11
	Art	1	4	1	3	2	1	1	0	13
	Business	0	0	1	1	0	0	0	2	4
	no major	0	0	1	1	2	2	2	1	9
Primary OS	Win. XP	1	2	3	1	3	2	3	3	18
	Win. 7	3	1	2	2	1	2	2	3	16
	Win. Vista	0	2	0	1	1	1	0	0	5
	Mac	2	0	1	1	1	1	1	0	7
	Linux	0	1	0	1	0	0	0	0	2
Computer usage (hrs/day)	Mean	5.67	5.33	4.67	6.33	5.67	5.33	5.67	6.50	5.64
	Range	1-10	2-10	2-10	3-12	3-9	1-9	3-10	3-10	1-12
Computer experience (years)	Mean	13.83	13	19.17	15.5	14.17	15.83	13.50	19.83	15.60
	Range	8-22	4-25	10-40	9-30	5-20	7-25	6-20	10-32	4-40

Table 4.8: Participants' demographics for each presentation order

4.4.4 Participants

We assigned 6 participants to each order of prompts presentation (8 orders) and recruited 48 participants from both the university and general community. Since all participants observed all 8 prompts, we could collect enough data for investigating participants' response behavior and information utilization. Also, the purpose of having 8 orders and counterbalancing the prompts presentation was not to find differences between different orders. Our purpose was to distribute the fatigue and learning impacts evenly across all presentation orders.

For recruitment, we sent out messages to email lists of several UBC departments, posted messages to Craigslist and Kijiji, and pinned flyers to community bulletin boards.

Since our pilot study showed that the participants' level of computer knowledge may have an impact on their behavior in the study, we balanced the level of computer knowledge of participants in each presentation order. The counter balancing was done to understand our participants' comprehension and response to the prompts in relation to their level of computer knowledge and expertise. For this purpose, we assigned 2 participants with high, 2 with medium and 2 with low level of computer knowledge to each presentation order. Therefore, we recruited 16 participants with high, 16 with medium and 16 with low level of computer knowledge for the study.

We used the following procedure for the recruitment:

When a subject responded to our recruitment message, we asked her to fill an online questionnaire about her demographics and computer knowledge. This questionnaire is shown in Appendix B.1. Based on participant' responses, we classified her computer knowledge level as low, medium or high. If the participant was eligible for participating in the study (we still needed participants with her level of computer knowledge and demographics) we scheduled a time for the study. In some cases, we refined our classification after participants performed the study tasks because their responses to online questionnaire did not match their behavior and answers in the user study. For example, there were participants whose responses showed that they have a high level of computer knowledge, but they were not able to explain the meaning of some information items such as program extension.

All participants were paid $15 CAD for their participation. Table 4.8 shows the demographics of our participants. They had a wide range of educational levels (from high school to Phd) and different majors. Also, the 24 non-student participants had a variety of occupations such as teachers, secretaries, managers, photographers, caterer and journalist or housewife and retired.

4.4.5 Analysis

We collected both qualitative and quantitative data in our study. For analyzing the quantitative data, we used proper statistical test such as ANOVA and χ^2 test; when χ^2 test's assumptions were not met (i.e., cells had an expected count < 5), we used the Fisher's exact test. To analyze the qualitative data, we used a card sorting approach [19]. We did not use qualitative methods such as grounded theory because our objective was to categorize the participants' behavior and knowledge. Also, since extracting a theory or high level patterns was not the goal of our research, we analyzed the data using card sorting approach. For this purpose, we first wrote our participants' responses to the interview questions on index cards. Then, we iteratively sorted the index cards for each question into multiple piles so that cards representing similar responses were in the same pile. We then associated a theme with each pile, that represented participants' understandings, misunderstandings, risks perception and preference for the information content of the prompt. This process was done iteratively to classify the responses.

4.5 Results

Our results include (1) participants' understanding of the prompt and each information item, (2) participants' risk perception and intended action in response to prompts (3) participants' information utilization (4) impact of participants' computer knowledge on information utilization (5) and participants' preference for information items.

4.5.1 Prompt understanding

In this section, we report our participants' initial understanding of the prompt purpose and each information item of the prompt.

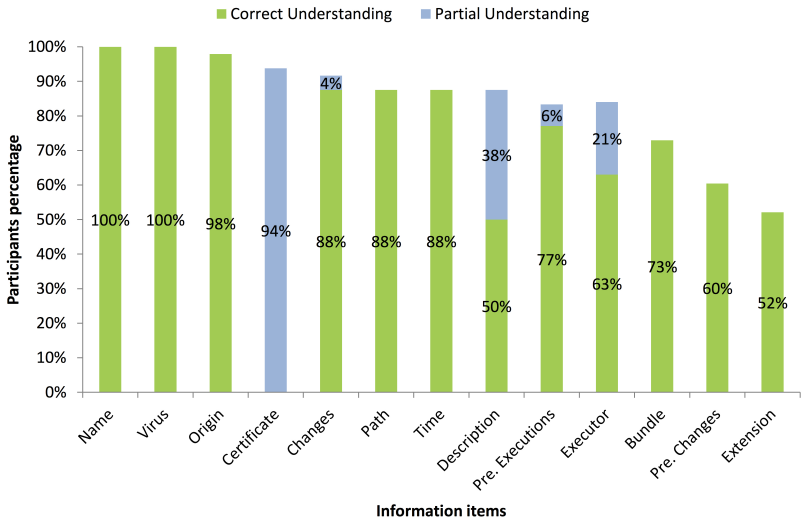


Figure 4.1: Percentage of participants who had correct or partially correct understanding of each information item

Prompt purpose

In our exit interview, we asked participants what the purpose of such prompt would be if they receive it on their own computer.

All participants except two had an almost correct understanding about the purpose of the prompt. Thirty two participants mentioned the purpose of the prompt is to ask their permission for applying changes on their computer or running the indicated program. Nine mentioned it provides them with information about the program or changes the program is going to apply and asks their decision about running this program. Two mentioned it intends to protect them from threats and three indicated it alerts them about the changes and risks associated with running the program.

Only two participants did not know the purpose of the prompt and could not provide a correct explanation about it.

Information item	Definitions acceptable for correct understanding	Definitions acceptable for partial correct understanding
Name	Name, description or identification of the file, program	N/A
Extension	Type or format of the file	N/A
Path	Where the program is located or stored at the computer	N/A
Origin	Where the file is taken from	N/A
Time	When the file was created / downloaded on the computer	
Description	Information or description about the program from a reliable reference	Description about the program, a good feature for the program
Changes	Consequences or changes the program applies	The files that are modified
Pre. changes	Changes the program has applied if it was executed before	N/A
Executor	The source executing, triggering or initiating the program	User executing the program
Pre. executions	Number of times the program was executed before	Number of times the program was downloaded before
Bundle	Program to be executed after / with this program, Program triggered by this program	N/A
Certificate	Shows whether the program is digitally signed by a trusted authority	Shows whether the program is verified , a good feature for the program
Virus	Shows if the program has any virus	N/A

Table 4.9: Criteria for classifying participants' understanding of each information item

Information items

We asked our participants to explain their understanding of each information item in the prompt. For responding to this question, participants provided a brief description about all thirteen items. Based on their responses, we classified participants as having correct or partially correct understanding of each information item. Table 4.9 shows our criteria for classifying participants' understanding of each information item. Since the "Digital certification" item in all prompts only indicated

whether program has a valid certificate or not, and the source which the certificate is issued to, we narrowed our criteria for judging participants' understanding of this item to the one mentioned in table 4.9.

To increase the reliability of our categorization, two researchers independently rated the participants' understandings of items. An inter-rater reliability analysis using the Kappa statistic was performed to determine consistency between the raters. The reliability was found to be Kappa = .796 ($p < .001$). While this shows a high agreement between raters, some participants were categorized differently. The two researchers subsequently discussed the categories with each other and achieved consensus on the categorization.

Figure 4.1 shows the percentage of participants who had a correct or partially correct understanding of each information item. As shown in the figure, most participants (above 80%) understood the name, virus, origin, certificate, changes, path, time, description, pre. executions and executor. However, extension, pre. changes and bundle were the items with the least understanding ratio (less than 75%). It should be noted that none of our participants had a correct understanding about the digital certification, however, most understood that having a valid certification increases the credibility of the program.

4.5.2 Risk perception and intended action

In this section, we report participants' response to each prompt and the level of risk that participants perceived from each prompt. Since the Fisher exact test indicated there was no statistically significant difference between responses of participants in different presentation orders, we combined the response of participants in different orders.

Risk perception

After participants responded to each prompt, they indicated their perceived level of risk from the prompt on a scale of 5 (very risky) to 1 (not risky). Figure 4.2 shows participants' perceived level of risk for each prompt. As shown in the figure, the perceived level of risk for the B_PreUsage prompt was higher than other benign prompts and even one malicious prompt. The reason is that participants paid more

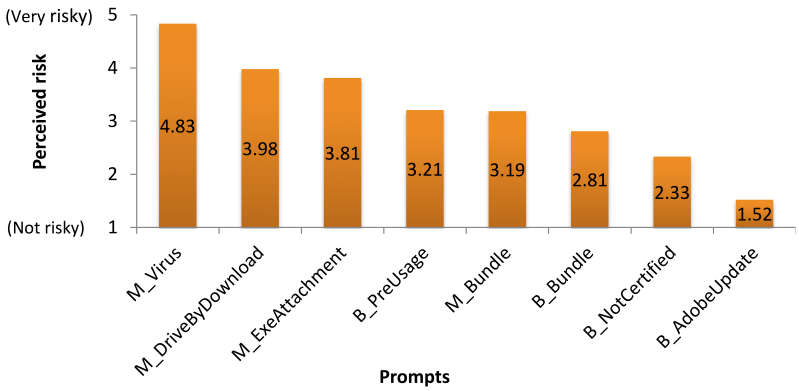


Figure 4.2: Level of risk perceived by participants for each prompt

attention to the changes that the program applies (access to firewall) than to the number of previous executions (executed 98 times). On average, they rated the impact of pre. executions as 1.15 and showed that this item encouraged them to continue the program while they rated the impact of changes as 2.42 and indicated this item encouraged them to stop the program.

Also, the perceived level of risk in the M_Bundle prompt was lower than other malicious prompts. The reason is that although on average participants indicated the bundle item encouraged them to stop the program, the mean impact was only 0.5 which was not strong enough to encourage participants to stop the program.

Response to prompts

All our participants responded to all the prompts. Figure 4.3 shows the percentage of participants who responded correctly to each prompt. As shown in the table, the correct responses to the prompts M_Bundle, B_PreUsage and B_Bundle were less than other prompts. These prompts were mainly intended to evaluate the use of "Bundle" , "Pre. changes" and "Pre. executions" fields. The low number of correct responses to these prompts shows that participants could not use these information items correctly.

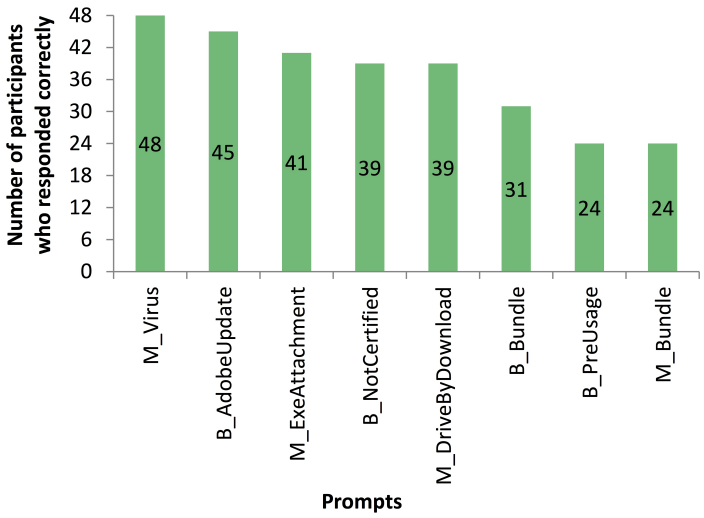


Figure 4.3: Percentage of participants who responded correctly to each prompt

4.5.3 Information utilization

After participants responded to each prompt, they indicated the usefulness of each item for making decision on a scale of 5 (very useful) to 1 (not useful). They also could indicate the fact that they did not understand or did not pay attention to an item. Table 4.10 shows the mean of usefulness rating for each item in every prompt. We sorted the rows based on the mean usefulness rating of each item in all prompts. Then we sorted the columns based on the mean usefulness rating of all items in each prompt.

We used participants' usefulness rating to identify the items which were useful for participants to make decision in each prompt. For this purpose, we used the following procedure for each prompt:

1. Conducted a one way ANOVA to compare the usefulness rating of items in a prompt.

2. Identified the most useful item: The most useful items is the item that has the highest mean of usefulness rating in the prompt.

3. Identified other useful items: These are the items that meet the following

conditions:

(a) There is no statistically significant difference between their mean of usefulness rating and the mean of usefulness rating of the most useful item.

(b) There is a statistically significant difference between their mean of usefulness rating and the mean of usefulness rating of at least one other item in the prompt.

The useful items, which are identified by the above procedure, are colored with green or red in table 4.10. To choose the color we used the "impact rating" provided by participants. After responding to each prompt, participants indicated the impact of each item on their decision (whether the item encouraged them to continue or stop the program) as well as the strength of this impact on a scale of 5 (very strong impact) to 0 (no impact). Table 4.11 shows the mean impact of each information item in each prompt. We represent the impact of items with positive or negative values depending on whether the item encouraged participants to continue or stop the program.

We colored the useful items in table 4.10 and 4.11 with green or red depending on whether their mean impact rating matched with the correct response to the prompt or not. The match and mismatch are defined as follows:

1. Match(Green):
   - Benign prompt: The mean impact rating of an item indicated that the item encouraged participants to continue the program.
   - Malicious prompt: The mean impact rating of an item indicated that the item encouraged participants to stop the program.
2. Mismatch(Red):
   - Benign prompt: The mean impact rating of an item indicated that the item encouraged participants to stop the program.
   - Malicious prompt: The mean impact rating of an item indicated that the item encouraged participants to continue the program.

Information Item	B_AdobeUpdate	B_PreUsage	M_DriveByDownload	B_Bundle	B_NotCertified	M_Bundle	M_ExeAttachment	M_Virus
Virus	4.53	3.69	3.30	3.98	4.29	3.77	3.35	4.90
Changes	3.02	3.98	4.34	3.13	3.36	2.81	4.49	2.83
Description	4.23	2.91	3.14	2.83	4.07	3.31	3.11	3.28
Origin	4.21	3.17	3.21	3.04	3.71	3.04	2.85	2.81
Certificate	4.57	2.91	3.17	3.18	2.67	3.43	3.07	2.83
Name	4.33	3.17	3.33	3.25	2.90	3.02	3.00	2.72
Executor	3.60	2.89	3.55	3.52	2.73	2.63	2.30	2.09
Extension	2.75	2.62	3.15	2.53	2.26	2.45	3.32	2.66
Path	3.24	3.16	2.93	2.33	2.60	2.49	2.26	2.43
Time	3.30	3.23	2.38	2.51	2.58	2.26	2.11	1.96
Bundle	2.41	2.24	2.09	3.74	2.52	2.83	1.87	1.87
Pre. executions	2.42	4.11	2.00	2.04	2.11	2.07	2.26	2.13
Pre. Changes	2.39	3.87	1.63	2.05	2.13	2.02	2.05	1.93

Table 4.10: Mean usefulness of each information item in each prompt. The most useful items are colored with green or red depending on whether their mean impact rating matched (green) or mismatched (red) with the correct response to the prompt.

Using table 4.10 and 4.11, we discuss how useful each information item was for participants in different prompts.

1. Virus: The "Virus" item was a useful item in all the benign prompts and encouraged participants to continue the program. It was also useful in one of the malicious prompts in which the program had a virus and this field strongly encouraged participants to stop the program. However, it also was useful in another malicious prompt (M_Bundle) and encouraged participants to continue the program. Since participants could not use the "Bundle" item correctly in this scenario, they used other items such as "Virus" and "Certificate" to respond to the prompt. Based on this result, the "Virus" item can be useful if there is a virus in the program or if no virus is found in the program

Information Item	B_AdobeUpdate	B_NotCertified	B_Bundle	M_Bundle	M_ExeAttachment	M_Virus	B_PreUsage	M_DriveByDownload
Virus	4.13	3.94	3.77	3.40	2.94	-4.48	3.08	3.02
Name	3.67	0.21	2.02	0.44	-0.52	0.69	-0.02	0.17
Time	2.17	1.17	1.30	0.56	0.33	0.00	0.90	0.04
Executor	2.60	1.46	1.23	0.83	0.71	1.06	1.31	-2.79
Origin	3.54	2.60	1.25	-0.25	0.56	-0.06	-0.73	-0.60
Bundle	1.23	1.40	1.61	-0.50	0.41	0.40	0.56	0.27
Path	2.25	1.31	1.35	0.96	0.60	0.36	-1.56	-0.67
Pre. Executions	0.83	0.58	0.50	0.08	0.23	0.04	1.15	-0.17
Pre. changes	0.96	0.56	0.50	0.38	-0.02	0.19	-1.42	0.23
Certificate	4.08	-1.90	-1.96	-2.17	1.32	2.35	-2.02	1.08
Extension	0.98	-0.81	0.60	-0.43	-1.81	0.21	-0.50	-1.06
Description	3.54	2.98	-1.79	-2.06	-2.15	-1.77	-1.98	-2.27
Changes	1.77	1.27	0.90	0.33	-3.75	-0.21	-2.42	-3.50

Table 4.11: Mean impact of each information item in each prompt. The most useful items are colored with green or red depending on whether their mean impact rating matched (green) or mismatched (red) with the correct response to the prompt.

and other items support the legitimacy of the program. For example, if a program has certification, description, or applies normal changes, it is beneficial for users to highlight the result by anti-virus. However, if other items are concerning, for example, the program has no description, no certification or applies untypical changes, it is better to hide this item to encourage the user to think carefully before making decision.

2. Changes: The "Changes" item was useful in four cases. In three of these cases, the program applied some untypical changes such as writing in "System" directory or accessing the firewall. Such changes concerned participants and encouraged them to stop the program. However, when the program applied typical changes such as writing in the "Program Files" directory,

participants did not find the “Changes” item very useful. It was only in the B_Bundle prompt that the program applied typical changes and participants found the “Changes” a useful item. Similar to the M_Bundle prompt, this prompt intended to investigate whether participants can use the “Bundle” item correctly. In this case, the “Bundle” item in the first prompt indicated the execution of the program in the second prompt. As many participants could not take benefit of the “Bundle” item correctly, they relied on other items such as “Changes” and “Certificate” to respond to the prompt. Therefore, “Changes” was useful when it differed from changes applied by most programs or when other items were not useful enough for making decisions.

3. Description: The “Description” item was useful in three cases. In two cases, B_AutoUpdate and B_NotCertified, the program had a description which encouraged participants to continue the program. It was also useful in the M_Bundle prompt in which the lack of description encouraged participants to stop the program. As mentioned before, the M_Bundle prompt was mainly intended to investigate whether participants can utilize the “Bundle” item correctly. Since most participants did not use this item correctly, they used other items such as “Description”, “Certificate” and “Origin” to respond to the prompt. This may show that lack of description is useful for deciding to stop the program. However, in other malicious prompts which participants could respond to the prompts correctly using other items, the lack of description was not as important as in the M_Bundle prompt. Overall, “Description” was a useful item when it was available. Also, if participants could not make decision based on other items, “Description” was an item that many participants used.

4. Origin: The “Origin” item was useful in three cases. In two of these cases (B_Adobe Update and B_NotCertified), the website was a well-known one. Therefore, when participants recognized the website, the origin was a useful information item which encouraged participants to continue the program. It was also useful in the M_Bundle prompt. This prompt was mainly intended to investigate whether participants can utilize the “Bundle” item correctly. In this prompt, participants were supposed to stop the program as the first

prompt in the bundling scenario did not indicate the execution of another program in the "Bundle" item. However, most participants did not use this item correctly and relied on other items such as "Origin", "Certificate" and "Description" to respond to the prompt. In this case, the origin was not a well-known website that encouraged participants to stop the program. This may show that an infamous origin is useful for deciding to stop the program. However, in other malicious prompts which participants could respond to the prompts correctly using other items, an infamous origin did not play a significant role. Overall, the "Origin" was more useful when participants recognized it. Also, if participants could not make decision based on other items, the origin was an item that many participants used.

5. Certificate: The "Certificate" item was useful in three cases. First, it was useful in the B_AdobeUpdate prompt in which the program had a valid certification and this item encouraged participants to continue the program. In three other benign prompts, the program did not have a certification. In two of these cases, the lack of digital certificate did not play a significant role in encouraging participants to stop the program. However, in the B_Bundle prompt, since many participants could not use the "Bundle" item correctly, they relied on other items such as "Certificate" and "Changes" to respond to the prompt. In this case, lacking a digital certificate encouraged some participants to stop the program. "Certificate" was also useful in the M_Bundle prompt in which the program did not have any certificate and this item encouraged participants to stop the program. As explained before, participants were expected to use the "Bundle" item in the M_Bundle prompt. However, since they could not use this item correctly, they used other items such as "Certificate". In three other malicious prompts, the program had a certificate but users found other items more useful to decide about the program legitimacy. Overall, "Certificate" was a useful item if other items did not have a strong impact on participants' decision.

6. Name: The "Name" item was useful in two prompts. First, it was useful in the B_AdobeUpdate prompt in which the program name was "AdobeReaderAutoUpdate". Since many participants recognized "Adobe", the name was

helpful for them in making decision. Second, the name was useful in the B_Bundle prompt. Since the name of the program in this prompt appeared in the first prompt of the bundling scenario, participants found the name a useful item for making decision. However, the "Name" was not a useful item when the name of the program was not well-known for participants. As shown in table 4.11, it neither encouraged participants to stop nor to continue the program.

7. Executor: The "Executor" item was useful in two cases. First, it was useful in the M_DriveByDownload prompt in which the content of "Executer" item was "Unknown". Second, it was useful in the "B_Bundle" prompt in which the content of "Executer" item was the name of the program in the first prompt. In other cases, the executor was the user or the operating system. Therefore, we can conclude the "Executor" item was more useful if it was unknown or it was a source other than a known source such as the user or the operating system.

8. Extension: Table 4.10 shows the "Extension" item was not useful in any prompt. However, when we identified the useful items for those participants who understood the information items correctly or partially correctly, we found that "Extension" was a useful item in the M_ExeAttachment and M_DriveBy Download prompts. In these two prompts, an executable file was going to run while the user expected a different file format. Therefore, the "Extension" can be useful if participants understand what the extension is. Our further analysis showed participants who could take advantage of the "Extension" item were those with a high level of computer knowledge. Section 4.5.3 and 4.5.4 provide more details about the impact of understanding items and participants' computer knowledge on the information utilization.

9. Path: The "Path" was not a useful information item in any prompt, however, its mean usefulness rating was higher in three prompts than other prompts. First, its usefulness rating was higher in the B_AdobeUpdate in which the path was the directory where Adobe was installed. Second and third, it was useful in the B_PreUsage and M_DriveByDownload prompts in which the

program was executed from a “Temp” directory. In other cases, the path was mainly “Downloads” directory. Therefore, if the path was different from the locations that participants usually work with, they paid more attention to it.

10. Time: The “Time” item was only useful in the B_PreUsage prompt. In this prompt, the creation time was a year ago. Also, the “Time” usefulness rating was almost high in the B_AdobeUpdate prompt in which the creation time was a week ago and matched with the installation time of Adobe according to the scenario. In other prompts, the creation time was about seconds or minutes before. Therefore, when the creation time was close to the current time, participants did not pay attention to it carefully. However, a creation time that was in the past grabbed their attention more.

11. Bundle: The “Bundle” item was useful in the B_Bundle prompt in which it showed the name of the program which triggered the second prompt. However, it was not useful in the M_Bundle prompt in which its content was “None” in the first prompt and a second prompt followed the first one. Therefore, this field was useful when it was other than “None”. When it was “None”, participants did not use it.

12. Pre. executions: The “Pre. Executions” item was useful in the B_PreUsage prompt in which this item showed the program was executed 98 times previously and the user continued the program all the 98 times. In this prompt, the “Pre. Executions” was a useful item which encouraged participants to continue the program. However, in other prompts that the content of this was zero, participants did not find it a useful item. Therefore, this item was useful if its content was not zero.

13. Pre. changes: This item was useful in “B_PreUsage” prompt. However, although we believed it should encourage participants to continue the program (because the “Changes” and “Pre. Changes” were the same which shows such changes were applied previously on the computer), it encouraged most participants to stop the program. Therefore, this item was not understood and used correctly by most of the participants.

Impact of participants' information understanding on information utilization

To investigate how understanding the information items impacts the information utilization, we identified the useful items for those participants who understood information items correctly or partially correctly. For this purpose, we used the same procedure mentioned in section 4.5.3, but for each item, we only considered the ratings of participants who understood the item correctly or partially correctly.

As a result, we obtained a table similar to table 4.10, but with few differences:

- Although "Extension" was not a useful item when we considered the ratings of all participants, it became a useful item in two prompts (M_ExeAttachment and M_DriveByDownload) when we considered the ratings of those who understood items. In these two prompts, an executable file was going to run while the user expected a different file format.
- Although, "Changes" and "Certificate" were useful items in the B_Bundle prompt when we considered the ratings of all participants, they were not useful anymore when we only considered the rating of those who understood the items. This was because these participants rated the usefulness of the "Bundle" item higher than all participants. Therefore, "Changes" and "Certificate" were not significant items for these participants.

"Extension", "Bundle" and "Pre. Changes" were the items with least ratio of understanding. When we only considered the usefulness rating of participants who understood the items, "Extension" became a useful item in two cases and "Bundle" became more useful in the B_Bundle prompt. However, the "Bundle" was not still useful in the M_Bundle prompt in which the content of "Bundle" item was 'None". It shows even though participants understood the "Bundle" field, they could not use it correctly when the content of this item was "None". Also, the "Pre. Changes" still encouraged participants to stop the program. Therefore, despite understanding the item, the participants could not use it correctly.

4.5.4 Impact of participants' computer knowledge and background on information utilization

To investigate how participants' level of computer knowledge impacts their response to prompts and information utilization, we conducted two other statistical tests:

1. Computer knowledge level impact on the response: We conducted a χ^2 test to compare the response of participants with different levels of computer knowledge to each prompt. The results showed that the number of correct responses of participants with high level of computer knowledge to the B_Bundle prompt was significantly higher than the correct response of participants with low level of computer knowledge ($\chi^2(1,32) = 6.78, p < .05$). Also, the number of correct response of participants with medium level of knowledge to the B_PreUsage prompt was significantly higher than the correct response of participants with low level of knowledge ($\chi^2(1,32) = 6.14, p < .05$).

2. Computer knowledge level impact on the usefulness rating: We conducted a one way ANOVA to compare the usefulness ratings of participants with different levels of computer knowledge in each prompt. Table 4.12 shows the items that their ratings were statistically significant different by participants with different levels of computer knowledge ($p < .05$). The last column shows the two knowledge levels that their participants had different ratings. The "Higher" rating column shows the level whose mean rating was higher. The "F" column is the value of F in ANOVA test.

   As shown in the table, the "Name" had higher usefulness rating by participants with high level of computer knowledge in the B_Bundle prompt as these participants noticed that the name of the program in the second prompt had appeared in the first prompt. They also paid more attention to the "Bundle" item in the B_Bundle prompt and noticed that it indicates the execution of the second program. The Origin was also more useful in the B_NotCertified for participants with high or medium level of computer knowledge as they recognized the "Cnet" website. Finally, the "Extension"

Information item	Prompt	Computer knowledge levels with statistically significant different usefulness rating		
		Higher rating	Lower rating	F
Extension	B_Bundle	High	Low	3.27
	M_ExeAttachment	High	Low	19.40
		High	Medium	19.40
		Medium	Low	19.40
	M_DriveByDownload	High	Low	5.09
	M_Virus	High	Low	5.31
	M_Bundle	High	Low	0.99
Bundle	B_Bundle	High	Low	4.57
		Medium	Low	4.57
Origin	B_NotCertified	High	Medium	
		Medium	Low	4.49
Name	B_Bundle	High	Low	5.47

Table 4.12: Information items that participants with different levels of computer knowledge rated their usefulness differently

was a useful item in many prompts for participants with high or medium level of computer knowledge.

The above results suggest that the "Extension", "Bundle" and "Origin" fields can be more useful for participants with high or medium level of computer knowledge.

We also investigated how participants' occupation impacted their response to prompts and information utilization. For this purpose, we conducted two other statistical tests:

1. Impact of occupation on the response: We conducted a χ^2 test to compare the response of student and non-student participants to each prompt. Since in some cases the cells had an expected count <5, we used the Fisher's exact test. The results showed that there is no statistically significance difference between the response of student and non-student participants in each prompt ($p > 0.05$ in each prompt).

2. Impact of occupation on the usefulness rating: We conducted a t-test to compare the usefulness ratings of student and non-student participants for each

Information item	Prompt
Path	B_NotCertified B_PreUsage M_ExeAttachment
Description	B_PreUsage
Certificate	B_PreUsage
Extension	B_ExeAttachment

Table 4.13: Information items that student and non-student participants rated their usefulness differently

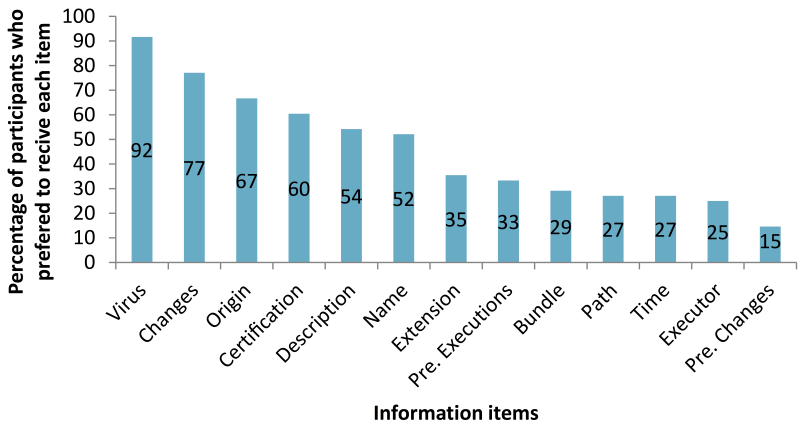


Figure 4.4: Percentage of participants who preferred to receive each item

item in each prompt. As a result we obtained 104 *Sig* values (8 prompts $*$ 13 items). Only six *Sig* values were less than 0.05 which their corresponding items and prompts are shown in table 4.13. The other 98 *Sig* values were greater than 0.05 and did not show any statistically significance difference between the ratings of student and non-student participants.

4.5.5 Information preference

During the exit interview, we asked participants which information items they prefer to receive in such prompts. Figure 4.4 shows the percentage of participants that preferred to receive each item. As the table shows, “Virus”, “Changes”, “Origin”, “Certificate”, “Description” and “Name” were chosen by more participants.

We also asked participants if they preferred other information items to be included in such prompts. Thirty three participants did not ask for any other information item. Other mentioned some information items such as a recommendation on the prompt (5), other users' comments and experience in using the program (3), program size (1), program version (1), virus type (1), anti-virus update time (1), name of the trusted source (1), program hash value (1) and more description on the changes (1). Such information can be provided in the detail section of the prompt.

The last question in our exit interview was to ask participants about their preference for receiving a descriptive or a short message from the operating system.

Twenty eight participants preferred to receive a descriptive message similar to the message they observed in the study. Nine found the current operating system messages not enough, but preferred a briefer message than the messages they observed in our study. Three wanted to receive some of the information items in the detail section of the prompt and 5 wanted to see a subset of information based on the context. Also, 4 participants preferred to receive a short message.

4.6 Discussion

Our analysis and discussion reveal how participants understood and utilized the thirteen information items in different contexts. In the following, we summarize our results by identifying the most understandable, useful and preferable information items. We also discuss a set of recommendations for designing a UAC prompt.

4.6.1 Most understandable, useful and preferable information items

Table 4.14 provides a summary of our results provided in section 4.5:

- In the first row (understanding), the items that were understood correctly or partially correctly by more than 80% of participants, are colored green.
- In the second row (usefulness), the items that their mean of usefulness rating were more than 3, are colored green.
- In the third row (preference), the items that were preferred by more than 50% of participants, are colored green.

	Name	Extension	Path	Origin	Time	Description	Changes	Pre. Changes	Executor	Pre. Executions	Bundle	Certificate	Virus
Understanding	■		■	■	■	■	■		■	■		■	■
Usefulness	■			■		■	■					■	■
Preference	■			■		■	■					■	■

Table 4.14: Most understandable, useful and preferable information items

As shown in the table, the name, origin, description, changes, certification and virus were the most understandable, useful and preferred items.

4.6.2 Design recommendations

Our results showed some information items were more useful for participants in specific contexts. Therefore, we recommend operating system designers to develop a context-based UAC prompt. This prompt selects different items to show in the main part of the prompt in each context. Other items can be shown in the detail section of the prompt. The other alternative is to show all the items but highlight a subset of them for the user. Such context-based presentation of items, makes the prompt polymorphic. A polymorphic prompt is more resistant to habituation and receives more attention from users [47] [48].

The guidelines below show how to select a subset of information items in each context.

1. Name: Although the "Name" was useful for participants to make decisions only when it was a well-known name or was referred in another prompt, it is inevitable to show it in every prompt.

2. Extension: This item is only useful for users with high or medium level of computer knowledge.

3. Path: The "Path" can be highlighted when it differs from the locations that users frequently work with or are familiar with.

4. Origin: The “Origin” can be useful if the user recognizes the origin. Therefore, we suggest showing a reputation measure for web sites in the prompt to help users decide about the website reputation.

5. Time: The “Time” can be highlighted when the creation time is not close to the current time.

6. Description: The “Description” is a useful piece of information. Although, it is more useful when it is available, we recommend to show its availability all the time as it was one the items with high usefulness rating in all the prompts, did not cause a dangerous behavior by participants and they used it when other information items were not enough for them to make decision.

7. Changes: “Changes” should be highlighted when the changes that the program applies are not similar to the changes that most programs apply.

8. Pre. changes: “Pre. changes” was not understood and used by participants correctly, therefore, it should be removed from the prompt.

9. Executor: The “Executor” should be highlighted when the executor is not known to the user. For example, when the executor is the user or the operating system, users do not pay attention to this field, however, if the executor is unknown or it is another program, users find this field more useful.

10. Pre. executions: “Pre. executions” can be highlighted when the number of previous executions is not zero. If it is zero, it will not be useful for users.

11. Bundle: The “Bundle” is mostly useful for users with high or medium level of computer knowledge. Also, these users will use the item correctly if it is not “None”.

12. Certificate: The “Certificate” is an overall useful item to be highlighted in every prompt and users use it when other items are not enough to make decision.

13. Virus: Although the “Virus” was a useful item in most of the prompts, we recommend to highlight this item if a virus is found in the program or if no

virus is found and other items such as description and certification support the legitimacy of the program. If these items do not encourage participants to continue the program, it is more beneficial for users to hide the result of anti-virus as there may be some risk associated with running the program and the fact that the program seems clean of virus, encourages participants to continue the program despite possible risks.

Based on the above design recommendation, an initial prototype for UAC prompt is designed and shown in figure 4.5 and figure 4.6. Figure 4.5 shows the minimal presentation of information items. Figure 4.6 shows the prompt when the user clicks on the "Show details" in the previous prompt. The user can click on the underlined items to view more details about them.

4.7 Limitations

The goal of our study was to identify the information items that assist users in assessing the risk of privilege elevation and responding to UAC prompts. We did not focus on improving the risk perception by users as operating systems can not estimate the risk of privilege elevation accurately. In some security warnings such as SSL warnings, the operating system has precise information (mismatch in the URL or expired certification) to estimate the risk with high accuracy. Given our study design and data collection protocol, there are some limitations and threats to validity of our results.

We made some assumptions, which are discussed in section 4.3.1, for selecting the information items of the UAC prompt. To implement these assumptions, some of the current mechanisms of the operating system should be modified or enhanced. Also, we assumed that the information items provided by the operating system can not be faked by the attacker. Although we selected the values of items based on Windows operating system, these items can be collected in other operating systems; however, the value of items will be different.

There are several internal threats to validity of our results. First, we did not change the locations of items in the prompt during the experiment. All par-

ticipants observed the items in the same location in all prompts. Therefore, participants' information utilization may have been influenced by the prompt design. For example, it is probable that participants payed less attention to items at the bottom of the prompt compared to items at the top of the prompt. Also, participants may have behaved differently as time passed due to the fatigue and learning issues. To decrease this threat, we counter-balanced the presentation order of prompts.

Moreover, there are some external threats to validity of our results. It is very challenging to recruit a sample that represents the real users' population. We balanced the demographics of our participants to decrease this threat. However, since we recruited equal number of participants with high, medium and low level of computer knowledge, our sample does not represent the real users' population. We made this choice to be able to compare the information utilization behavior of participants with different levels of computer expertise. We also conducted our study in the lab environment. Therefore, it is probable that participants were not motivated to respond to prompts as they do in their normal computer usage. Also, they may not be motivated to provide accurate ratings in the post-task questionnaires. To motivate participants, we informed them in the beginning of the study that a prize would be given to participants who pay attention to items and provide accurate ratings.

4.8 Conclusion

A UAC prompt can be a layer of defense against many types of malware. However, users do not respond to current operating systems UAC prompts correctly. One reason is that the information content of such prompts is not useful for users to assess the risk of privilege elevation correctly. To address this problem, we considered thirteen different information items to be included on a UAC prompt so that users can assess the risk of privilege elevation more accurately. These items were mostly selected based on the results of our previous study. Our user study with 48 participants showed that the program name, origin, description, certification, changes to apply and result of scan by anti-virus are the most understandable, useful and preferred

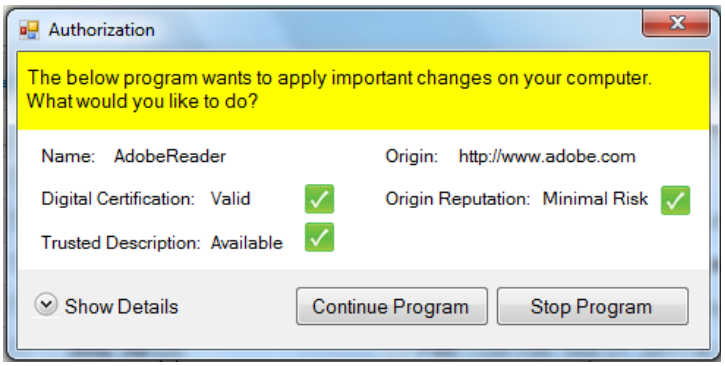


Figure 4.5: Initial prototype of context-based UAC prompt - Minimal presentation of information items

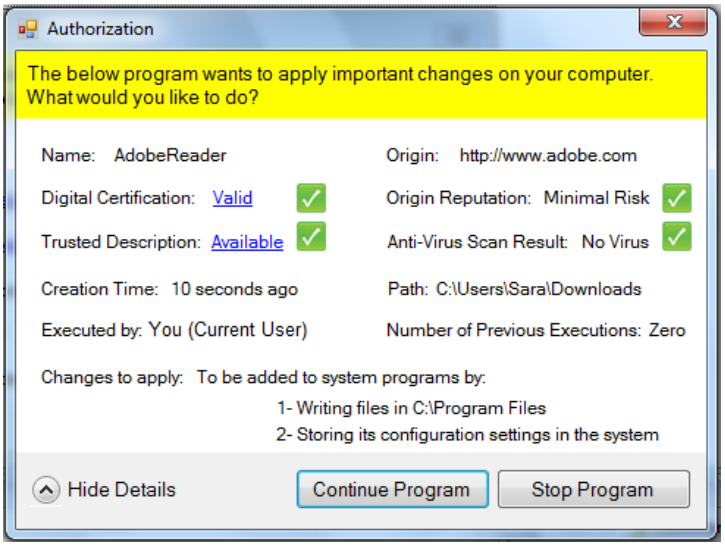


Figure 4.6: Initial prototype of context-based UAC prompt - Showing all details for the program

items for users. To avoid habituation, decrease cognitive load on users and improve their response to the prompts, we recommend changing the UAC prompt to a context-based prompt which presents different items to users based on the context. We have provided a set of guidelines for selecting the appropriate information items in different contexts.

Chapter 5

Conclusion

In this thesis, our goal was to investigate how well main-stream operating systems for personal computers support users in following the principle of least privilege, or PLP for short, and what can be done to improve such support. For this purpose, we first studied the users' understanding, behavior and challenges in using two practical implementations of PLP in Windows Vista and 7, named low-privileged user accounts and user account control (UAC) prompts. Our research scope was narrowed to Windows Vista and Windows 7 as these two main operating systems are currently used by most of the users [41]. Based on our findings, we provided guidelines for improving one of the implementations of PLP, UAC prompts.

In the following, we provide a summary of the findings and contributions of each two parts of our research.

- **Investigating User Account Control Practices**: A user study and contextual interview with 30 Windows Vista and 15 Windows 7 participants revealed these users' understanding, behavior and challenges in using low-privileged user accounts and user account control prompts.

  All these participants used an admin account on their laptop for performing daily computer tasks as they were not aware about the high risks of using high-privileged accounts and benefits of using low-privileged accounts. Although, many participants could explain and show in practice that the low-privileged accounts are sufficient for performing daily tasks, they were not

motivated to use these accounts. In addition to lack of understanding about the security aspects of these accounts, prior experience with the inconvenience of using low-privileged accounts discouraged our participants from using such accounts. Therefore, we recommend operating system designers to convey the benefits of low-privileged accounts and risks of high-privileged accounts to users. They also should provide users with default low-privileged accounts and encourage users to use these accounts by providing a means for applying legitimate modifications on their computer conveniently with in low-privileged accounts.

Our results also showed 69% of users do not use the UAC approach correctly. Lack of understanding about the purpose of UAC prompts and the risks it aims to mitigate was the main reason of poor performance of participants in using this implementation of PLP. Also, when participants were in the context of doing an action, they thought any UAC prompt is related to their current action and consented to the privilege elevation. Based on our findings, we recommend operating system designers to communicate the purpose of UAC prompts and enough information for making decision about privilege elevation to users, integrate the UAC prompt with other security-related mechanisms and raise UAC prompts in fewer situations.

- **Information Content for Assessing the Risk in Privilege Elevation**: We provided guidelines for improving the content of UAC prompt since such prompt can be a layer of defense against different types of malware. Also, if these prompts are responded correctly, the PLP will be followed without sacrificing the convenience of using high-privileged accounts. Our first study showed that users are able to respond correctly to UAC prompts and prefer to receive these prompts if they understand the purpose of UAC and the risks it intends to mitigate. We focused on the content of the UAC prompt to be able to distinguish the effect of content from presentation and help users to assess the risk of privilege elevation more accurately. For this purpose, we included thirteen different information items on the prompt mostly based on the results of our first study. Our user study with 48 participants showed it is more beneficial for users to change the UAC prompt to a context-based

prompt which presents a subset of information items to user in each context. We have provided guidelines that determine which items should be presented to users in a specific context. Among the thirteen information items, the program name, program origin, program description from a trusted source, changes the program applies, program digital certification and the result of program scan by anti-virus were the items that were understood, used and preferred by most of the participants.

While the focus of this thesis was on Windows Vista and 7, we believe that our research can be used in other operating systems design, especially those that implement the principle of least privilege by using privilege elevation prompts.

5.1 Future work

There are several directions for the future research.

- Understanding user account control practices in other operating systems: Currently our data about users' understanding and behavior in using the implementations of PLP, is limited to Windows Vista and 7 users. A follow up to our research is to extend our study to other operating system users. One option is to study a larger number of participants by conducting a survey to obtain information about users' knowledge and usage patterns of user accounts and privilege elevation prompts. While the number of user studies is limited, surveys can provide a large number of responses. By using the user study result, better survey questions can be designed. Also, comparing the results of the survey with the user study findings can determine which aspects of our findings might be generalizable to a larger population.

  The other option is to conduct a similar user study with users of other operating systems such as Linux and Mac OS X. In order to be able to compare the results of these user studies, their protocol and design should be similar as much as possible.

- Implementing our recommendations for improving the UAC and LUA:

  We have offered several recommendations for improving the UAC and LUA approaches. Implementing each of this recommendation can be a direction

for the future research. For example, one can investigate how to decrease the number of UAC prompts or integrate such prompts with other operating system security-related mechanisms. Also, investigating approaches that make working with low-privileged accounts convenient for users and inform them about the benefits of these accounts is an important and challenging direction for the future research.

- Design, implementation and evaluation of context-based UAC prompts:

  We provided a set of guidelines for designing a context-based UAC prompt. The next step is to evaluate our guidelines. For this purpose, the same study (explained in chapter 4) can be performed using a context-based prompt that presents the items based on our guidelines. The items should be presented in the same format as they were presented in our study so that we can distinguish the effect of presentation from the effect of content. If the result of this study confirms the result of our study (explained in chapter 4), then an interface designer can proceed with designing the prompt interface. However, the presentation of items using icons and text should not invalidate the prior studies results. For this purpose, the interface should be evaluated and revised in multiple pilot studies so that users have the same understanding and risk perception of the items as in prior studies. The last step is to conduct a naturalistic longitudinal study to evaluate how users respond to the context-based UAC prompt and how users' information utilization changes in the real context of use. Such naturalistic longitudinal can be performed using other types of UAC prompts (Smart UAC and Norton UAC). Therefore, the response behavior of participants to each type of UAC prompt can be compared.

Bibliography

[1] Application Manifests. Application Manifests. http://msdn.microsoft.com/en-us/library/aa374191%28v=vs.85%29.aspx, 2010. URL http://msdn.microsoft.com/en-us/library/aa374191%28v=vs.85%29.aspx.

[2] A. Besmer, J. Watson, and H. R. Lipford. The impact of social navigation on privacy policy configuration. In *SOUPS '10: Proceedings of the Sixth Symposium on Usable Privacy and Security*, pages 1–10, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0264-7. doi:http://doi.acm.org/10.1145/1837110.1837120.

[3] C. Bravo-Lillo, L. Cranor, J. Downs, and S. Komanduri. Poster: What is still wrong with security warnings: a mental models approach. In *SOUPS '10: Proceedings of the 6th Symposium on Usable Privacy and Security*, New York, NY, USA, 2010. ACM.

[4] J. C. Brustoloni and R. Villamarín-Salomón. Improving security decisions with polymorphic and audited dialogs. In *SOUPS '07: Proceedings of the 3rd symposium on Usable privacy and security*, pages 76–85, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-801-5. doi:http://doi.acm.org/10.1145/1280680.1280691.

[5] L. F. Cranor. A framework for reasoning about the human in the loop. In *UPSEC'08: Proceedings of the 1st Conference on Usability, Psychology, and Security*, pages 1–15, Berkeley, CA, USA, 2008. USENIX Association.

[6] P. Dourish, R. E. Grinter, J. D. de la Flor, and M. Joseph. Security in the wild: user strategies for managing security as an everyday, practical problem. *Personal and Ubiquitous Computing*, 8(6):391–401, 2004. ISSN 1617-4917.

[7] J. S. Downs, M. B. Holbrook, and L. F. Cranor. Decision strategies and susceptibility to phishing. In *SOUPS '06: Proceedings of the Second*

Symposium on Usable Privacy and Security, pages 79–90, New York, NY, USA, 2006. ACM. ISBN 1-59593-448-0. doi:http://doi.acm.org/10.1145/1143120.1143131.

[8] S. Egelman, J. King, R. C. Miller, N. Ragouzis, and E. Shehan. Security user studies: methodologies and best practices. In *CHI Extended Abstracts*, pages 2833–2836. ACM, 2007.

[9] S. Egelman, A. B. Brush, and K. M. Inkpen. Family accounts: a new paradigm for user accounts within the home environment. In *CSCW '08: Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, pages 669–678, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-007-4. doi:http://doi.acm.org/10.1145/1460563.1460666.

[10] S. Egelman, L. F. Cranor, and J. Hong. You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In *CHI '08: Proceedings of the SIGCHI Conference on Human factors in Computing Systems*, pages 1065–1074, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-011-1. doi:http://doi.acm.org/10.1145/1357054.1357219.

[11] J. Goecks, W. K. Edwards, and E. D. Mynatt. Challenges in supporting end-user privacy and security management with social navigation. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, SOUPS '09, pages 5:1–5:12, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-736-3. doi:http://doi.acm.org/10.1145/1572532.1572539. URL http://doi.acm.org/10.1145/1572532.1572539.

[12] C. Jensen and C. Potts. Privacy policies as decision-making tools: an evaluation of online privacy notices. In *CHI*, pages 471–478, 2004.

[13] S. Kim. Java web start: Developing and distributing Java applications for the client side. White paper, IBM Corp. Armonk, NY, September 2001. http://www.ibm.com/developerworks/java/library/j-webstart.

[14] S. Leonard, H. Otani, and M. Wogalter. Comprehension and memory. *Warnings and risk communication*, pages 149–187, 1999.

[15] Mac OS X Security. An introduction to Mac OS X security. http://developer.apple.com/internet/security/securityintro.html, August 2004.

[16] J. E. McGrath. Methodology matters: doing research in the behavioral and social sciences. *Human-computer interaction: toward the year 2000*, pages 152–169, 1995. Morgan Kaufmann Publishers Inc.

[17] G. McGraw and E. W. Felten. *Securing Java: Getting down to business with mobile code*. John Wiley & Sons, 2 edition, 1999.

[18] R. Naraine. Adobe PDF exploits using signed certificates, bypasses ASLR/DEP. http://www.zdnet.com/blog/security/adobe-pdf-exploits-using-signed-certificates-bypasses-aslrdep/7303, September 2010. URL http://www.zdnet.com/blog/security/adobe-pdf-exploits-using-signed-certificates-bypasses-aslrdep/7303.

[19] J. Nielsen. Card sorting to discover the users' model of the information space. http://www.useit.com/papers/sun/cardsort.html, 1995.

[20] Norton UAC. Vista User Account Control. http://us.norton.com/theme.jsp?themeid=labs_uac&header=0&depthpath=0.

[21] Page Hit Ranking. Page Hit Ranking. http://distrowatch.com, April 2011. URL http://distrowatch.com.

[22] W. Poole. Financial Analyst Meeting. http://www.microsoft.com/msft/speech/FY05/PooleFAM2005.mspx, July 2005. URL http://www.microsoft.com/msft/speech/FY05/PooleFAM2005.mspx.

[23] F. Raja, K. Hawkey, P. Jaferian, K. Beznosov, and K. S. Booth. It's Too Complicated, So I Turned It Off! Expectations, Perceptions, and Misconceptions of Personal Firewalls. In *Proceedings of the 3rd ACM Workshop on Assurable & Usable Security Configuration (SafeConfig)*, October 4 2010.

[24] Root Sudo. Rootsudo. Ubuntu Documentation, https://help.ubuntu.com/community/RootSudo, February 2010.

[25] C. A. Rusen. Windows 7 vs Windows Vista: the UAC Benchmark. http://www.7tutorials.com/windows-7-vs-windows-vista-uac-benchmark, August 2009.

[26] M. Russinovich. Inside Windows 7 User Account Control. TechNet Magazine, July 2009.

[27] J. Saltzer and M. Schroeder. The protection of information in computer systems. *Proceedings of the IEEE*, 63(9):1278–1308, Sept. 1975. ISSN 0018-9219.

[28] F. B. Schneider. Least privilege and more. *IEEE Security and Privacy*, 1: 55–59, September 2003. ISSN 1540-7993. doi:10.1109/MSECP.2003.1236236. URL http://dl.acm.org/citation.cfm?id=1435675.1436943.

[29] SELinux. SELinux. http://selinuxproject.org.

[30] Smart UAC. Smart UAC. http://www.replaceuac.com/.

[31] A. Sotirakopoulos, K. Hawkey, and K. Beznosov. "I did it because I trusted you": Challenges with the Study Environment Biasing Participant Behaviours. In *SOUPS Usable Security Experiment Reports (USER) Workshop*, 2010.

[32] A. Steven. Applying the principle of least privilege to user accounts on Windows XP. Microsoft TechNet Library, http://technet.microsoft.com/en-us/library/bb456992.aspx, 2006.

[33] D. W. Stewart and I. M. Martin. Intended and unintended consequences of warning messages: A review and synthesis of empirical research. *Journal of Public Policy and Marketing*, 13(1):1–19, 1994.

[34] M. Stiegler and M. Miller. A capability-based client: The DarpaBrowser. Technical report, Focused Research Topic 5. Combex, Inc., Meadowbrook, PA, June 2002.

[35] M. Stiegler, A. H. Karp, K.-P. Yee, T. Close, and M. S. Miller. Polaris: virus-safe computing for Windows XP. *Commun. ACM*, 49(9):83–88, 2006.

[36] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor. Crying Wolf: An empirical study of SSL warning effectiveness. In *Proceedings of 18th USENIX Security Symposium*, pages 399–432, 2009.

[37] SUSE AppArmor. SUSE AppArmor. http://www.suse.com/support/security/apparmor.

[38] UAC. How Windows Vista helps protect computers from malware. TechNet Library, http://technet.microsoft.com/en-us/library/cc507865.aspx, September 2006.

[39] UAC. Understanding and configuring user account control in Windows Vista. http://technet.microsoft.com/en-us/library/cc709628(WS.10).aspx, 2007.

[40] UAC. What's new in user account control. TechNet Library, http://technet.microsoft.com/en-us/library/dd446675(WS.10).aspx, June 2009.

[41] Usage share of operating systems. Usage share of operating systems. http://en.wikipedia.org/wiki/Usage_share_of_operating_systems, April 2011. URL http://en.wikipedia.org/wiki/Usage_share_of_operating_systems.

[42] R. M. Villamarín-Salomón and J. C. Brustoloni. Improving user decisions about opening potentially dangerous attachments in email clients. In *E-Mail Clients, Poster, Symposium on Usable Privacy and Security, CMU*, pages 06–136, 2006.

[43] Vista Security Web Page. Some guidelines for securing your Windows Vista PC. http://download.microsoft.com, Security_Best_Practice_Guidance_for_Consumers.doc, 2007.

[44] R. N. M. Watson, J. Anderson, B. Laurie, and K. Kennaway. Capsicum: practical capabilities for unix. In *Proceedings of the 19th USENIX conference on Security*, USENIX Security'10, pages 3–3, Berkeley, CA, USA, 2010. USENIX Association. ISBN 888-7-6666-5555-4. URL http://dl.acm.org/citation.cfm?id=1929820.1929824.

[45] M. Wogalter. Purpose and scope of warnings. In *Handbook of Warnings*, pages 3–9. Lawrence Erlbaum Associates, 2006.

[46] M. Wogalter. Communication-Human Information Processing (C-HIP) Model. In *Handbook of Warnings*, pages 51–61. Lawrence Erlbaum Associates, 2006.

[47] M. Wogalter and C. Mayhorn. The future of risk communication: Technology-based warning systems. In *Handbook of Warnings*, page 783ââ“793. Lawrence Erlbaum Associates, 2006.

[48] M. Wogalter, V. Conzola, and T. Smith-Jackson. Research-based guidelines for warning design and evaluation. *Applied Ergonomics*, 33(3):219–230, 2002.

[49] J. S. Wolff and M. S. Wogalter. Comprehension of pictorial symbols: Effects of context and test method. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 40:173–186(14), 1998.

[50] G. Wurster and P. C. Van Oorschot. System configuration as a privilege. In *Proceedings of the 4th USENIX conference on Hot topics in security*, HotSec'09, pages 12–12, Berkeley, CA, USA, 2009. USENIX Association. URL http://dl.acm.org/citation.cfm?id=1855628.1855640.

[51] G. Wurster and P. C. van Oorschot. A control point for reducing root abuse of file-system privileges. In *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, pages 224–236, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0245-6. doi:http://doi.acm.org/10.1145/1866307.1866333. URL http://doi.acm.org/10.1145/1866307.1866333.

[52] H. Xia and J. C. Brustoloni. Hardening web browsers against man-in-the-middle and eavesdropping attacks. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 489–498, New York, NY, USA, 2005. ACM. ISBN 1-59593-046-9. doi:http://doi.acm.org/10.1145/1060745.1060817.

[53] S. Young and D. Lovvoll. Intermediate processing stages: Methodological considerations for research on warnings. *Warnings and risk communication*, pages 27–52, 1999.

[54] M. E. Zurko, C. Kaufman, K. Spanbauer, and C. Bassett. Did you ever have to make up your mind? what notes users do when faced with a security decision. In *ACSAC '02: Proceedings of the 18th Annual Computer Security Applications Conference*, pages 371–381, Washington, DC, USA, 2002. IEEE Computer Society. ISBN 0-7695-1828-1.

Appendix A

First User Study Documents

A.1 Background questionnaire

Background questionnaire

Gender:	Age:
Last Educational Degree:	Educational Major:
Current Occupation:	

1. How many hours a day do you use a computer? (If you do not use computer on a daily basis, please specify the usage frequency in terms of number of hours you use computer per week or month.) ……
2. How long have you been using a computer? ……

3. What operating system do you use the most frequently currently?

A) Windows XP
B) Windows 7
C) Windows Vista
D) Mac OS
E) Linux
F) Other (Please Specify):

4. How long have you been using this operating system? …….

5. If you use more than one computer or more than one operating system, please complete the following table. You can write the Usage Frequency in terms of number of hours/day; hours/week or hours/month.

Computer ID	Laptop / PC	Operating Systems	Usage Frequency
Computer 1			
Computer 2			
Computer 3			

6. In the above table, please circle the row that represents the laptop you have brought to this session.

Note: Please answer the rest of questions based on the laptop you have brought to this session.

7. What is your purpose for using this computer? (Select all that apply)

A) Multi Media (Music, Video, Photo)
B) Games
C) Word Processing
D) Spread Sheets
E) Presentation
F) Web surfing
G) Email
H) Instant Message
I) Reading News
J) Online Shopping
K) Online Gaming
L) Online Education
M) Pay Bills
N) Banking
O) Research
P) Programming
Q) Database applications
R) Other: …….

8. Who installed the operating system on this laptop?

A) Myself
B) A computer technician
C) Do not know
D) It was installed when I bought the computer
E) other. Please specify: ……

9. What security software do you have on this laptop? (Check all that apply)

A) Anti Virus
B) Windows Firewall
C) Any firewall other than Windows Firewall
D) Encryption Software
E) Spyware removal tool
F) Password manager
G) Others:----------
H) Do not know

10. How often you update your anti virus software?

A) It is done automatically.
B) Never
C) My antivirus license has been expired.
D) ……. time(s) per …… (week, month, year)
E) Somebody else does it: ……

11. How often do you perform a full system anti virus scan?

A) It is done automatically.
B) Never
C) My antivirus license has been expired.
D) ……. time(s) per …… (week, month, year)
E) Somebody else does it: ……

12. Please rate how difficult you find each of the following tasks to be.

Task	Very Easy	Somewhat easy	Neutral	Somewhat difficult	Very Difficult	Do not know	Never Done
Copying and moving files between directories							
Loading new software onto a computer							
Finding required information or services in Internet							
Installing operating system							
Administering a computer network server							
Writing a computer program							

13. Do other people use your computer?
A) Yes
B) No

If you answered "yes", please specify the following information about the people that use your computer.

Your Relationship to person	Usage purpose	Usage Frequency
……	……	……
……	……	……

A.2 Task instruction

Tasks instruction

Note: You need connectivity to Internet for performing the below tasks. Please use the following information when you start your browser:
User name: motiee
Password: (will be provided by the investigator)

Task 1-1- DVD player

You have received a DVD from one of your friends. Your current media player cannot play the file and you need a "DVD player" application to watch the DVD. You are interested to watch the file as soon as possible.
What do you do usually in such a situation?

- Do you buy the software from a computer store?
- Do you download free software?
- Do you buy the software online?
- Do you ask a friend to get the software from?
- Do you take another approach?

If you usually download free software, please perform the same steps now. Please try to think aloud while you do the task.
If you do not usually download free software, please let the investigator know.

Task 1-2- Editor

Your friend has given you a Text Editor application on his USB which is named "Text Hawk". He has recommended to you to install the application. What do you usually do in such situation? Please perform the same steps here.
The application installation file is given to you by the investigator.

If you have done the similar task before, please perform the same steps here. Please try to think aloud while you do the task. If you have any question or need any assistance for performing this task, please ask the investigator.

Task 1-3- Spyware Blaster
Your friend who is a computer security expert has suggested that "Spyware Blaster" is a suitable utility that blocks all kinds of spyware that get through your web browser. Download and install this application on your computer.

If you have done the similar task before, please perform the same steps here. Please try to think aloud while you do the task. If you have any question or need any assistance for performing this task, please ask the investigator.

Task 2- User account creation

Your brother would like to use your laptop. He mainly is going to check his emails, browse Internet, read documents and use Microsoft office applications. Create a user account for him in your system so that he can use your laptop.
Please try to think aloud while you do the task. If you need any question or assistance for performing this task, please ask the investigator.

A.3 Interview questions

Interview Questions

After performing the first experiment, the actions that participants performed during the experiment will be reviewed and they will be asked the following questions about each action:

Questions about Installation task

1- What kind of criteria do you consider when you download and install application from Internet?
2- Do you pay attention to warnings and messages that you get from operating system or browser when you download and install from Internet?
3- Do you know what is a UAC prompt?
4- (I showed the participant the UAC prompt she got when she installed Camtasia) Have you seen similar to these prompts before?
5- Do you know why do you get these prompts? What is the purpose of these prompts?
6- How do you respond to these prompts?
7- In what situations do you get these prompts?
8- (I showed them one instance of yellow prompt, whether the fake update or a yellow prompt they got in task1). Do you think this prompt is different from the previous one? Have you noticed the difference before? If yes, what is the difference and do you respond differently?
9- (I showed them the yellow update cache prompt) Do you remember you got this prompt? Why did you confirm or cancel the prompt?
10- How many prompts do you think you should get when you install an application?
11- Do you remember how many prompts you got when you installed text editor?
12- (If said yes to question 10), Why did you confirm both prompts?
13- Did the prompts interfere with your task performance or are they annoying? Why?
14- Do you know the prompts can be disabled?
15- If you knew the prompts can be disabled, would you disable them?

Note: If user does not do the installation, the following questions will be asked:

1- Why did not you do the installation? What are your concerns?
2- If you have to download and install an application in the future, what will you do?
3- Does anybody do this task for you?

Questions about Account Creation task

1- Before starting the task, I asked whether she knew the difference between administrative and non-administrative account? She fills also a table. I also ask whether she knows her account type.
2- (When the participants wanted to create the user account, she saw the administrative UAC, after she responded), Do you know why did you get this

prompt? Do you think it is different from the previous ones? Did the prompt interfere with your task performance?

3- Why did you created a standard / administrative account?
4- In what situations you may create another type of accounts?
5- Why you use administrative account to log in to your computer?
6- Have you created an account before? What type? Why?
7- What type of account have you used on your previous systems?
8- Have you ever worked with non-admin account? Where? Did you face any problem?

Appendix B

Second User Study Documents

B.1 Online background questionnaire

Online Background Questionnaire

1. Please fill in the following information.

   Name

   Gender

   Age

   Last degree

   Major

   Current occupation

   Are you a student?

   email

2. What do you use your computer for? (Check all that apply)
   - ☐ Multi Media (Music, Video, Photo)
   - ☐ Games
   - ☐ Word Processing
   - ☐ Spread Sheets
   - ☐ Presentation
   - ☐ Web surfing
   - ☐ Email
   - ☐ Instant Message
   - ☐ Online Shopping
   - ☐ Online Gaming
   - ☐ Online Education
   - ☐ Pay Bills
   - ☐ Banking
   - ☐ Research
   - ☐ Programming
   - ☐ Database applications
   - ☐ Others:----------

3. How do you assess your technical knowledge of computers?
   - ☐ 5 - Advanced at operating system level
   - ☐ 4 - Basic knowledge at operating system level
   - ☐ 3 - Advanced user of basic programs (web browsers, email, etc.)
   - ☐ 2 - Regular user of basic programs (web browsers, email, etc.)
   - ☐ 1 - Sporadic user of basic programs (web browsers, email, etc.)

4. Please indicate how difficult you find each of the following tasks to be.

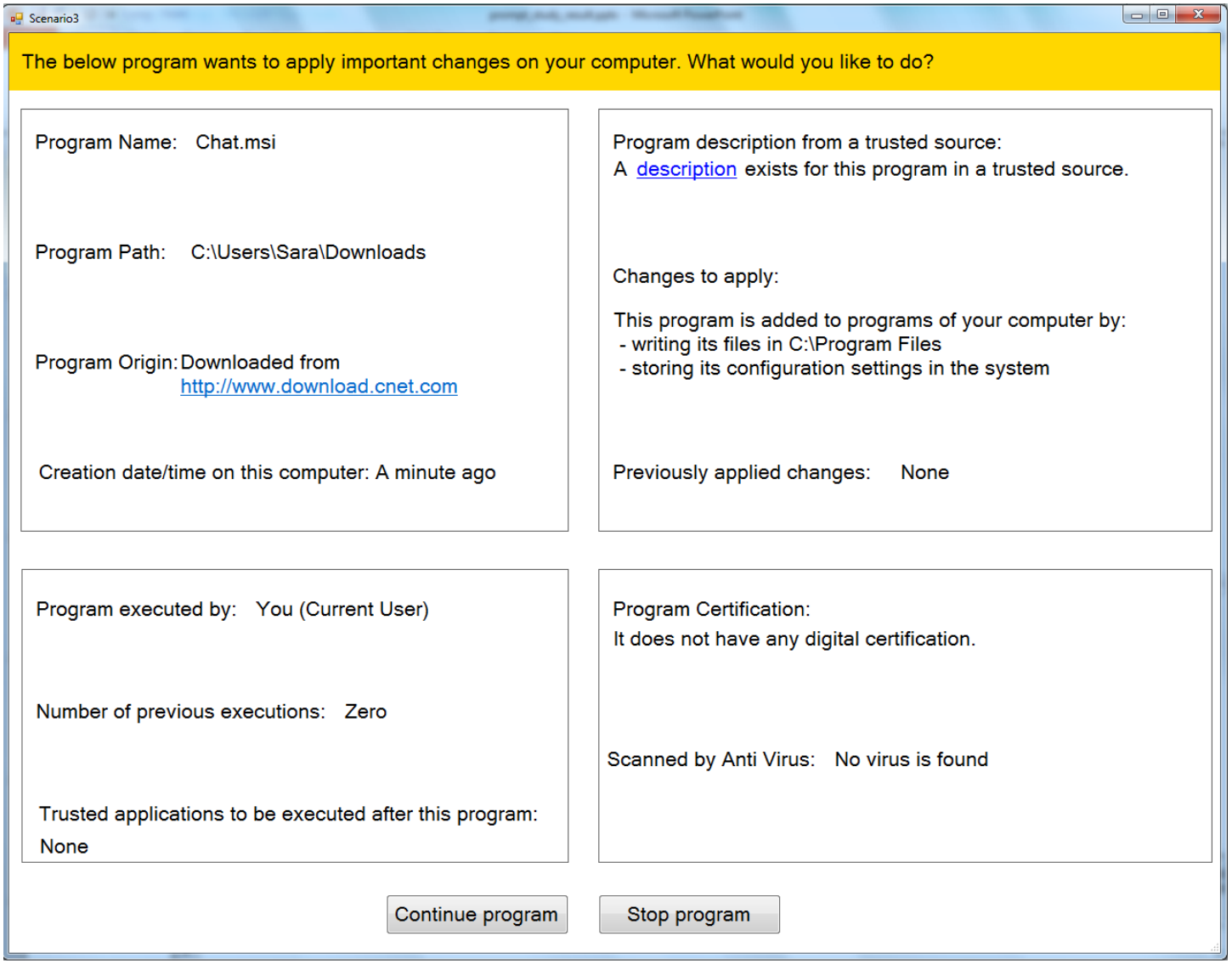
Task	Very Easy	Somewhat easy	Neutral	Somewhat difficult	Very Difficult	Do not know	Never Done
Copying and moving files between directories							
Finding required information or services in Internet							
Loading (installing) new software onto a computer							
Installing a device driver							
Installing operating system							
Computer programming							
Administering a computer network server							
Setting up a wireless network at home							

5. How do you troubleshoot your computer? (Select Always, often, sometimes or never for each of the below sentences.)

	Always	Often	Sometimes	Never
I troubleshoot my computer myself.				
I use system Help or online resources.				
I get help from other people.				

B.2 Task instruction and post-task questionnaire

Task and prompt sample:



You want to chat with your friend. You visit download.cnet.com and download a chat program which seems has interesting features. When you double click the installation file, you see the prompt shown on the screen now. Please read the prompt carefully and respond to it.

1- What is your response to the prompt? …….

2- Given a scale from (1-5), please indicate how useful each information was for you when deciding how to respond to the prompt?

Information item	Very Useful				Not Useful	I did not pay attention	I did not understand it
Program name	5	4	3	2	1	0	
Program extension	5	4	3	2	1	0	
Program path	5	4	3	2	1	0	
Program origin	5	4	3	2	1	0	
Creation date/time on this computer	5	4	3	2	1	0	
Program Description from a trusted source	5	4	3	2	1	0	
Changes to apply	5	4	3	2	1	0	
Previously applied changes	5	4	3	2	1	0	
Program executed by	5	4	3	2	1	0	
Number of previous executions	5	4	3	2	1	0	
Other applications to be executed	5	4	3	2	1	0	
Program Certification	5	4	3	2	1	0	
Scanned by Anti Virus	5	4	3	2	1	0	

3- What do you think about the hazardousness (risk of running the program) when you see this prompt?

Very Risky				Not risky at all
5	4	3	2	1

4- What was the individual impact of each information item on your decision? How strong was this impact?

Information item	Encourages to “Continue Program”	Encourages to “Stop Program”	No impact	Very Strong				Not Strong
Program name				5	4	3	2	1
Program extension				5	4	3	2	1
Program path				5	4	3	2	1
Program origin				5	4	3	2	1
Creation date/time on this computer				5	4	3	2	1
Program Description from a trusted source				5	4	3	2	1
Changes to apply				5	4	3	2	1
Previously applied changes				5	4	3	2	1
Program executed by				5	4	3	2	1
Number of previous executions				5	4	3	2	1
Other applications to be executed				5	4	3	2	1
Program Certification				5	4	3	2	1
Scanned by Anti Virus				5	4	3	2	1

B.3 Interview questions

Interview Questions

After performing the installation tasks, participants will be asked the following questions:

Questions about Installation task

1- What was the purpose of the prompts you observed in the study? Did all have a similar purpose or different ones?
2- What is the meaning of each information?
3- Which information was the most helpful information on the prompt?
4- Which information was not helpful and you think should be removed from the prompt?
5- Did you understand the question on the prompt?
6- Can you think of any other information that should be included on the prompt?
7- I will ask participants about their rationale for selecting specific ratings.