USABLE SECURITY

GRAD SEC

SEP 28 2017

USER AUTHENTICATION

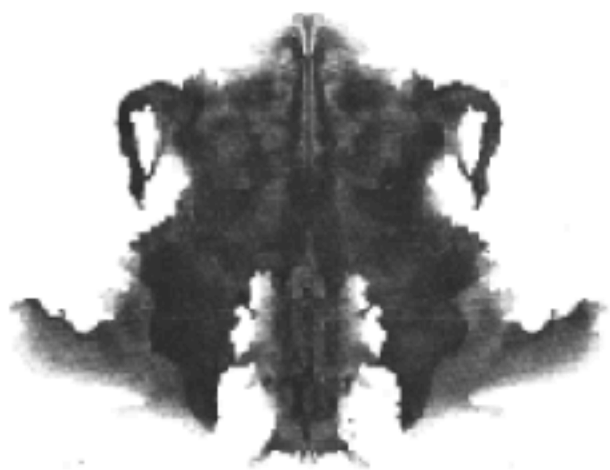
What we know (passwords)

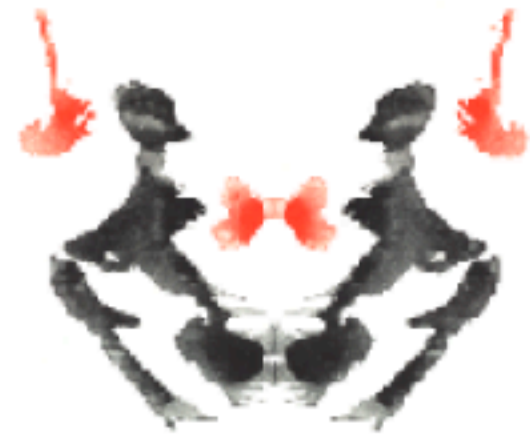
What we have (tokens)

What we are (iris, fingerprint)
[Accuracy vs. cost trade-off]

Other

USER AUTHENTICATION



INKBLOT AUTHENTICATION

Come up with two characters per image

DO WE NEED STRONG PASSWORDS?

What's the threat model?

How should we store passwords?

Is the attack online or offline?

Is the attack *targeted* or seeking *any user*?

DO WE NEED STRONG PASSWORDS?

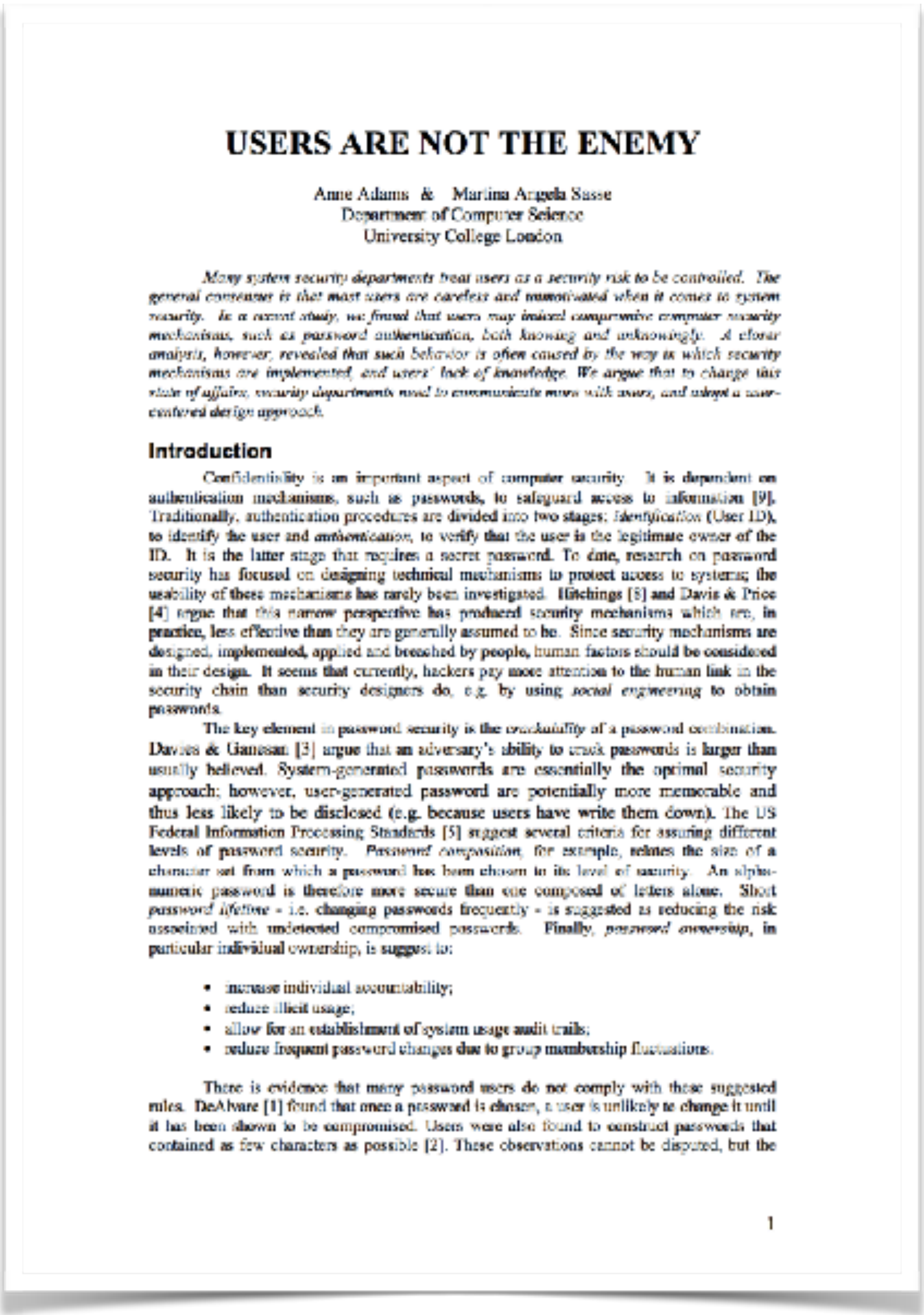
Let's consider offline attacks

6-digit passwords
+ 3-strikes-you're-out

Let's give the attacker 10 years to guess

10 years = ~10^4 passwords = ~1%

TODAY'S PAPERS



USERS ARE NOT THE ENEMY

Anne Adams & Martina Angela Sasse
Department of Computer Science
University College London

Many system security departments treat users as a security risk to be controlled. The general consensus is that most users are careless and unmotivated when it comes to system security. In a recent study, we found that users may indeed compromise computer security mechanisms, such as password authentication, both knowing and unknowingly. A closer analysis, however, revealed that such behavior is often caused by the way in which security mechanisms are implemented, and users' lack of knowledge. We argue that to change this state of affairs, security departments need to communicate more with users, and adopt a user-centered design approach.

Introduction

Confidentiality is an important aspect of computer security. It is dependent on authentication mechanisms, such as passwords, to safeguard access to information [9]. Traditionally, authentication procedures are divided into two stages: *Identification* (User ID), to identify the user and *authentication*, to verify that the user is the legitimate owner of the ID. It is the latter stage that requires a secret password. To date, research on password security has focused on designing technical mechanisms to protect access to systems; the usability of these mechanisms has rarely been investigated. Hitchings [8] and Davis & Price [4] argue that this narrow perspective has produced security mechanisms which are, in practice, less effective than they are generally assumed to be. Since security mechanisms are designed, implemented, applied and breached by people, human factors should be considered in their design. It seems that currently, hackers pay more attention to the human link in the security chain than security designers do, e.g. by using *social engineering* to obtain passwords.

The key element in password security is the *crackability* of a password combination. Davies & Ganesan [3] argue that an adversary's ability to crack passwords is larger than usually believed. System-generated passwords are essentially the optimal security approach; however, user-generated password are potentially more memorable and thus less likely to be disclosed (e.g. because users have write them down). The US Federal Information Processing Standards [5] suggest several criteria for assuring different levels of password security. *Password composition*, for example, relates the size of a character set from which a password has been chosen to its level of security. An alpha-numeric password is therefore more secure than one composed of letters alone. Short *password lifetime* - i.e. changing passwords frequently - is suggested as reducing the risk associated with undetected compromised passwords. Finally, *password ownership*, in particular individual ownership, is suggest to:

- increase individual accountability;
- reduce illicit usage;
- allow for an establishment of system usage audit trails;
- reduce frequent password changes due to group membership fluctuations.

There is evidence that many password users do not comply with these suggested rules. DeAlvare [1] found that once a password is chosen, a user is unlikely to change it until it has been shown to be compromised. Users were also found to construct passwords that contained as few characters as possible [2]. These observations cannot be disputed, but the

Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0

Alma Whitten
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
alma@cs.cmu.edu

J. D. Tygar[1]
EECS and SIMS
University of California
Berkeley, CA 94720
tygar@cs.berkeley.edu

Abstract

User errors cause or contribute to most computer security failures, yet user interfaces for security still tend to be clumsy, confusing, or near-nonexistent. Is this simply due to a failure to apply standard user interface design techniques to security? We argue that, on the contrary, effective security requires a different usability standard, and that it will not be achieved through the user interface design techniques appropriate to other types of consumer software.

To test this hypothesis, we performed a case study of a security program which does have a good user interface by general standards: PGP 5.0. Our case study used a cognitive walkthrough analysis together with a laboratory user test to evaluate whether PGP 5.0 can be successfully used by cryptography novices to achieve effective electronic mail security. The analysis found a number of user interface design flaws that may contribute to security failures, and the user test demonstrated that when our test participants were given 90 minutes in which to sign and encrypt a message using PGP 5.0, the majority of them were unable to do so successfully.

We conclude that PGP 5.0 is not usable enough to provide effective security for most computer users, despite its attractive graphical user interface, supporting our hypothesis that user interface design for effective security remains an open problem. We close with a brief description of our continuing work on the development and application of user interface design principles and techniques for security.

1 Introduction

Security mechanisms are only effective when used correctly. Strong cryptography, provably correct protocols, and bug-free code will not provide security if the people who use the software forget to click on the encrypt button when they need privacy, give up on a communication protocol because they are too confused about which cryptographic keys they need to use, or accidentally configure their access control mechanisms to make their private data world-readable. Problems such as these are already quite serious: at least one researcher [2] has claimed that configuration errors are the probable cause of more than 90% of all computer security failures. Since average citizens are now increasingly encouraged to make use of networked computers for private transactions, the need to make security manageable for even untrained users has become critical [4, 9].

This is inescapably a user interface design problem. Legal remedies, increased automation, and user training provide only limited solutions. Individual users may not have the resources to pursue an attacker legally, and may not even realize that an attack took place. Automation may work for securing a communications channel, but not for setting access control policy when a user wants to share some files and not others. Employees can be required to attend training sessions, but home computer users cannot.

Why, then, is there such a lack of good user interface design for security? Are existing general user interface design principles adequate for security? To answer these questions, we must first understand what kind of usability security requires in order to be

[1] Also at Computer Science Department, Carnegie Mellon University (on leave).

BONUS PAPER

Users Really Do Plug in USB Drives They Find

Matthew Tischer† Zakir Durumeric‡† Sam Foster† Sunny Duan†
Alec Mori† Elie Bursztein◇ Michael Bailey†

† University of Illinois, Urbana Champaign ‡ University of Michigan ◇ Google, Inc.
{tischer1, sfoster3, syduan2, ajmori2, mdbailey}@illinois.edu
zakir@umich.edu [illegible]@google.com

***Abstract*—We investigate the anecdotal belief that end users will pick up and plug in USB flash drives they find by completing a controlled experiment in which we drop 297 flash drives on a large university campus. We find that the attack is effective with an estimated success rate of 45–98% and expeditious with the first drive connected in less than six minutes. We analyze the types of drives users connected and survey those users to understand their motivation and security profile. We find that a drive's appearance does not increase attack success. Instead, users connect the drive with the altruistic intention of finding the owner. These individuals are not technically incompetent, but are rather typical community members who appear to take more recreational risks than their peers. We conclude with lessons learned and discussion on how social engineering attacks—while less technical—continue to be an effective attack vector that our community has yet to successfully address.**

I. Introduction

The security community has long held the belief that users can be socially engineered into picking up and plugging in seemingly lost USB flash drives they find. Unfortunately, whether driven by altruistic motives or human curiosity, the user unknowingly opens their organization to an internal attack when they connect the drive—a physical Trojan horse. Our community is filled with anecdotes of these attacks and pentesters have even boasted that they can *hack humans* by crafting labels that will pique an individual's curiosity [19]. "While in the bathroom, I place an envelope in one stall. On the cover of the envelope I put a sticker that says PRIVATE. Inside the 'private' envelope is a USB key with a malicious payload on it. I do this in one stall and also in the hallway by a break room to increase my chances and hope that the person that finds one of them is curious enough to insert it into their computer. Sure enough, this method seems to always work."

However, despite recent attacks that underscore the risk of malicious peripherals [39], [55] and rumors of the attack's efficacy, there has been little formal analysis of whether the attack is effective nor why users connect the drives. In this work, we investigate the classic anecdote by conducting a large scale experiment in which we drop nearly 300 flash drives of different types, in different locations, and at different times on the University of Illinois, Urbana-Champaign campus.

We measure the efficacy and speed of the attack by replacing expected files on the drive with HTML files containing an embedded img tag that allows us to track when a file is opened on each drive without automatically executing any code. We find that users pick up and connect an estimated 45%–98% of the drives we dropped. Further, the attack is expeditious with a median time to connection of 6.9 hours and the first connection occurring within six minutes from when the drive was dropped. Contrary to popular belief, the appearance of a drive does not increase the likelihood that someone will connect it to their computer. Instead, users connect all types of drives unless there are other means of locating the owner—suggesting that participants are altruistically motivated. However, while users initially connect the drive with altruistic intentions, nearly half are overcome with curiosity and open intriguing files—such as vacation photos—before trying to find the drive's owner.

To better understand users' motivations and rationale, we offered participants the opportunity to complete a short survey when they opened any of the files and read about the study. In this survey, we ask users why they connected the drive, the precautions they took, demographic information, as well as standard questions to measure their risk profile and computer expertise. We find that attack was effective against all subpopulations at Illinois. The majority of respondents connected a drive to locate its owner (68%) or out of curiosity (18%), although a handful also admitted they planned on keeping the drive for themselves.

The students and staff that connected the drives were not computer nor security illiterate and were not significantly different than their peers at the University of Illinois on Egelman and Peer's Security Behavior Intentions Scale (SeBIS) [12]. While the users that connected the drive engaged in riskier behavior than their peers on the DOSPERT scale [4], they were more risk averse than the general population in every domain except for recreational risk.

When prompted, 68% of users stated that they took no precautions when connecting the drive. For those respondents who considered protective measures, 10 (16%) scanned the drive with their anti-virus software and 5 (8%) believed that their operating system or security software would protect them, e.g., "I trust my macbook to be a good defense against viruses". Surprisingly, another 5 (8%) sacrificed a personal computer or used university resources to protect their personal equipment. In the end, all but a handful of the users who took precautions did so in an ineffective manner and the majority took no precautions at all.

These results—particularly the risk averseness relative to the general population on the DOSPERT scale—suggest that the attack would be effective against most users and that the average person does not understand the danger of connecting an unknown peripheral to their computer. We hope that by bringing these details to light, we remind the security community that

EXPERIMENT SETUP

297 USB drives dropped around campus

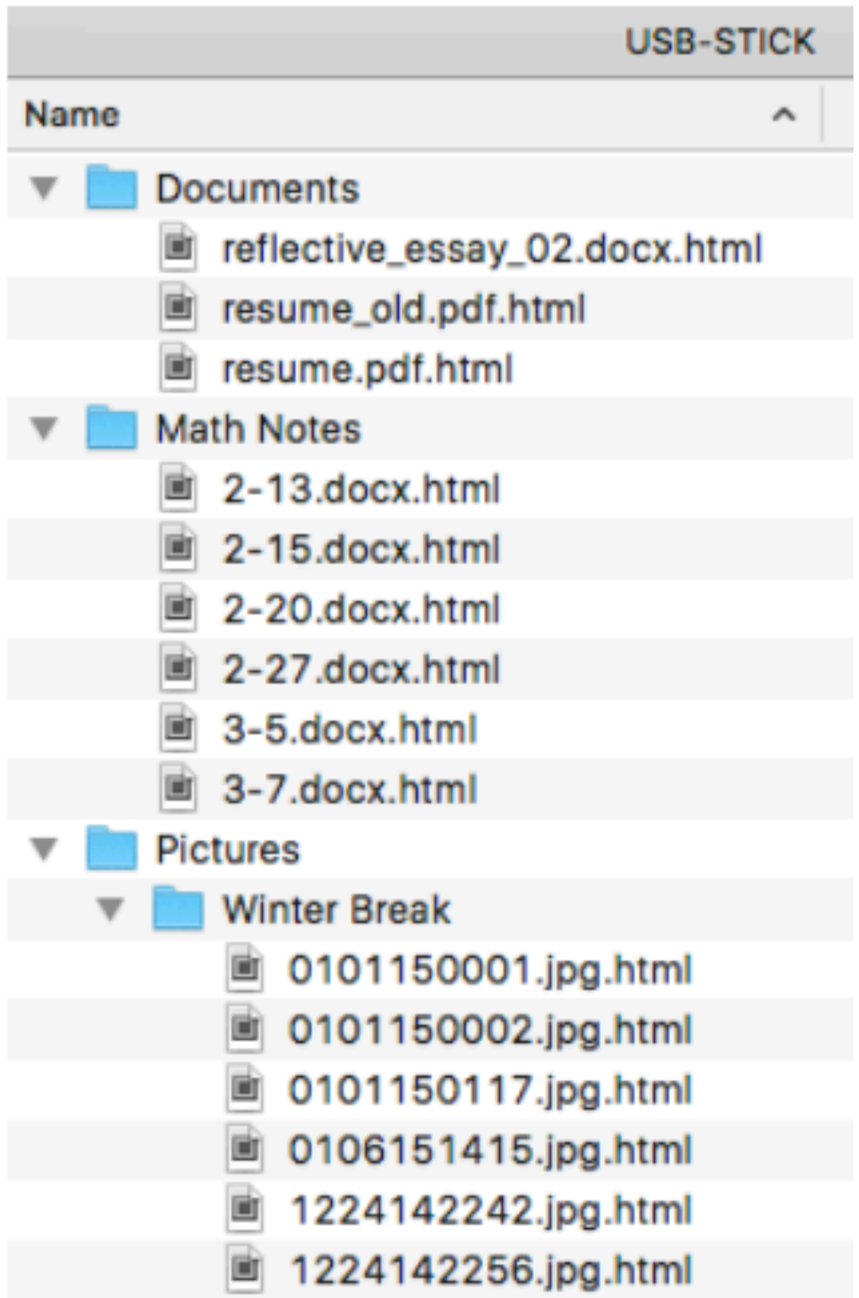
Varied location, time of day, and appearance:

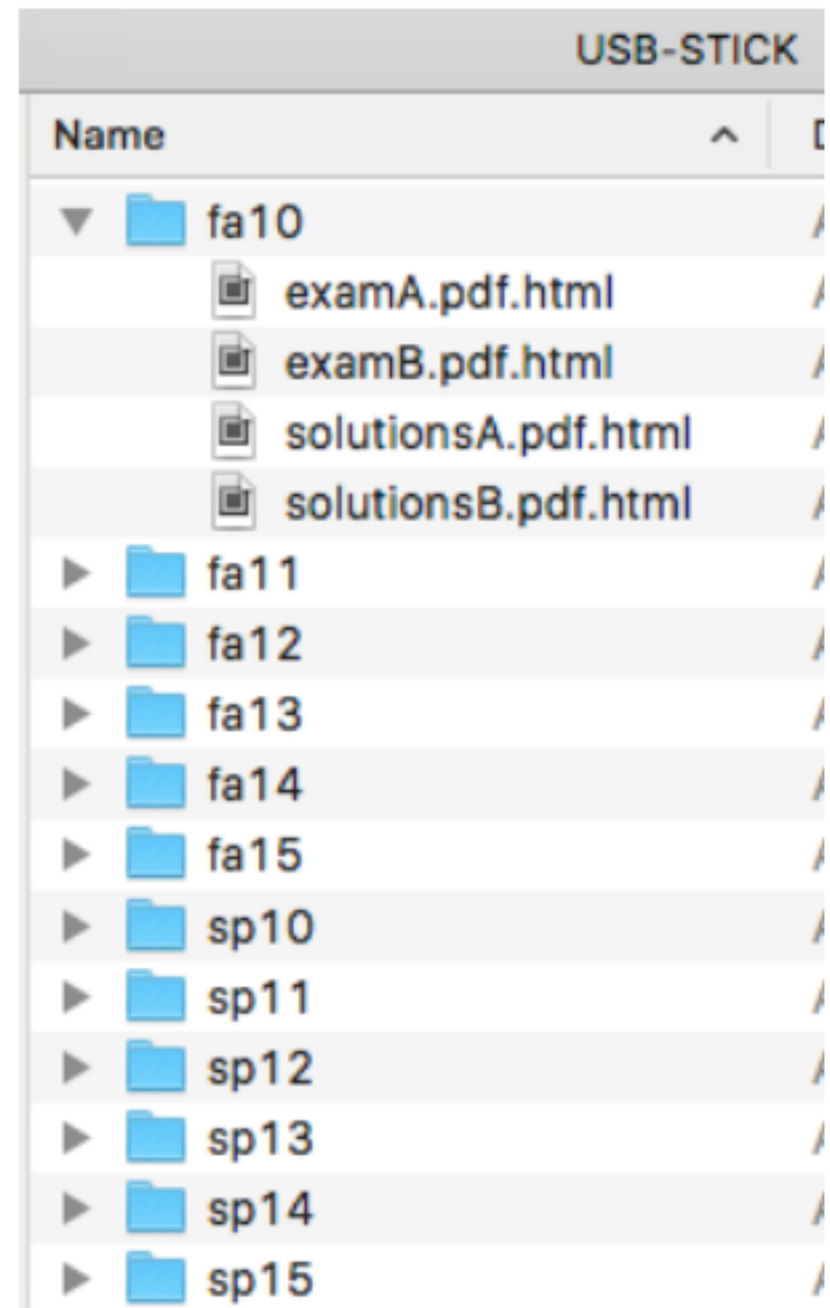
(a) Unlabeled drive (b) Drive with keys (c) Drive with return label (d) Confidential drive (e) Exam solutions drive

Fig. 1: **Drive Appearances**—We dropped five different types of drives. We chose two appearances (keys and return label) to motivate altruism and two appearances (confidential and exam solutions) to motivate self-interest, as well as an unlabeled control.

Periodically went to the locations to see what was taken/when

EXPERIMENT SETUP





All files are .html page informing them they're part of a study

<img> hits the measurement server + Survey

FINDINGS

Users Really Do Plug in USB Drives They Find

45% of the drives had a file open

98% of the drives were removed

Might have plugged it in but not opened a file

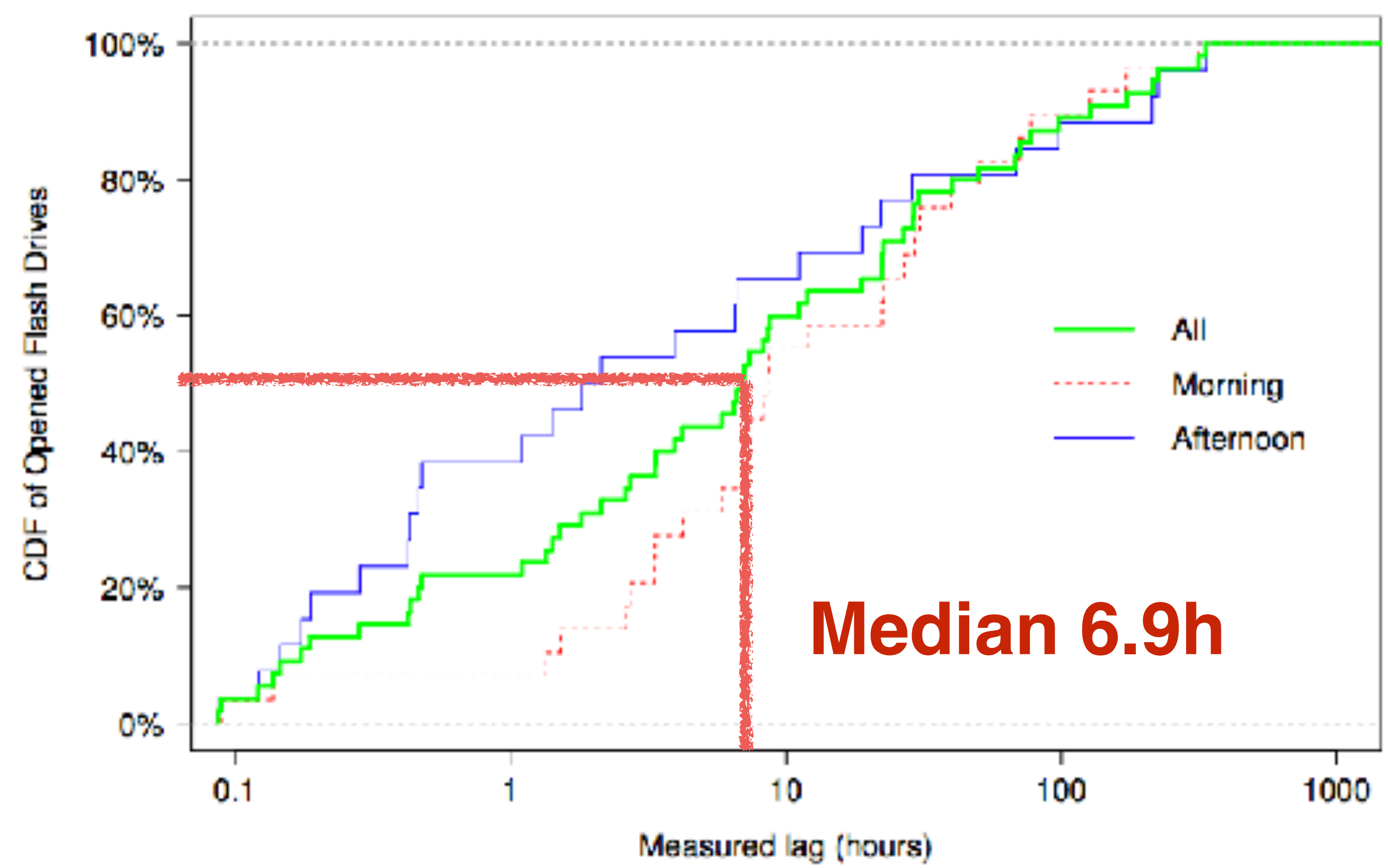


Fig. 3: **Empirical CDF of Measured Lag**—We show the empirical cumulative distribution function for the time difference between when a drive was dropped and when a file was opened on that drive. Afternoon drives were picked up more quickly than morning ones, but both were generally picked up quickly.

WHY DID THEY DO IT?

(a) Unlabeled drive (b) Drive with keys (c) Drive with return label (d) Confidential drive (e) Exam solutions drive

Fewer opened files

Perhaps they opened it altruistically to return it?

WHY DID THEY DO IT?

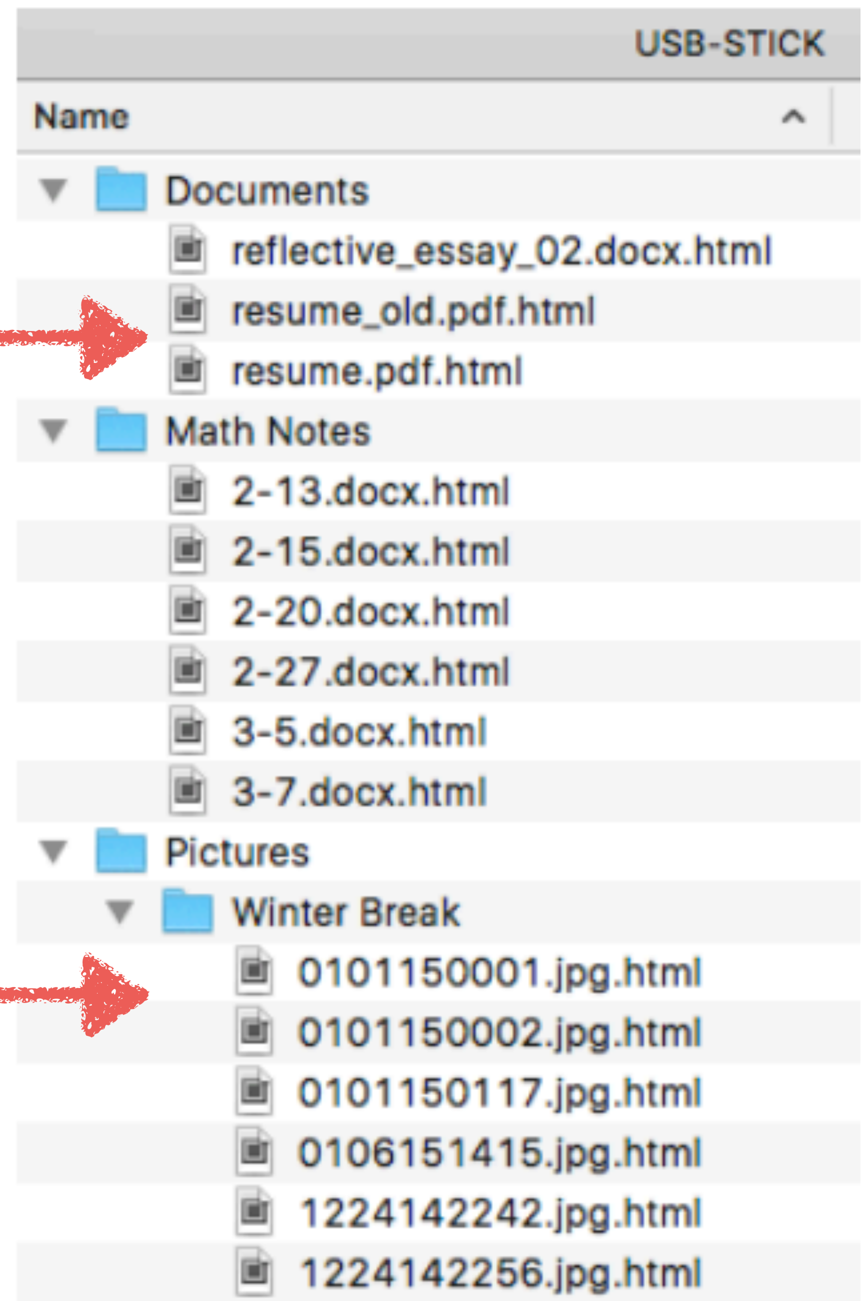
Code	Respondents	
Return drive	42	(68%)
Curious	11	(18%)
Listed location as response	5	(8%)
Keep drive	2	(3%)
Given drive by someone else	2	(3%)

TABLE V: **Participant Motivation**—We show the primary reasons given as responses to the question "Why did you pick up the flash drive and insert it into your computer?". Most respondents expressed a desire to return the flash drive, although many respondents also expressed curiosity.

WHY DID THEY DO IT?

Altruism?

Reality
(50% with
return label)



WHY TAKE THE RISK?

Code	Respondents	
Specific Precautions		
Scanned files with anti-virus	10	(16%)
Mentioned OS security features	5	(8%)
Sacrificed a computer	5	(8%)
Opened a file in a text editor	4	(6%)
Sandboxed a file	3	(5%)
Contacted/Web searched researcher	2	(3%)
Specific Words		
No	42	(68%)
Yes	8	(13%)

TABLE VI: **Participant Precautions**—We show coded responses to the question "Did you take any precautions before opening the file on the flash drive (e.g., scanning it for viruses)?". Most respondents did not take formal protection measures, although those that did employed a variety of methods.

WHY TAKE THE RISK?

Domain-specific risk taking (DOSPERT) scale

Test for risk aversion (higher = riskier), different categories

Domain-Specific Risk-Taking (Adult) Scale – Risk Taking

For each of the following statements, please indicate the **likelihood** that you would engage in the described activity or behavior if you were to find yourself in that situation. Provide a rating from *Extremely Unlikely* to *Extremely Likely*, using the following scale:

1	2	3	4	5	6	7
Extremely Unlikely	Moderately Unlikely	Somewhat Unlikely	Not Sure	Somewhat Likely	Moderately Likely	Extremely Likely

1. Admitting that your tastes are different from those of a friend. (S)
2. Going camping in the wilderness. (R)
3. Betting a day's income at the horse races. (F/G)
4. Investing 10% of your annual income in a moderate growth diversified fund. (F/I)
5. Drinking heavily at a social function. (H/S)
6. Taking some questionable deductions on your income tax return. (E)
7. Disagreeing with an authority figure on a major issue. (S)
8. Betting a day's income at a high-stake poker game. (F/G)
9. Having an affair with a married man/woman. (E)
10. Passing off somebody else's work as your own. (E)

WHY TAKE THE RISK?

Domain-specific risk taking (DOSPERT) scale

Test for risk aversion (higher = riskier), different categories

Risk Domain	Blais and Weber μ	Blais and Weber σ	USB μ	USB σ	t	df	p
Ethical	17.97	7.16	12.82	4.96	6.02	138.29	1.48E-08
Financial	20.67	8.51	15.32	5.22	0.67	157.94	7.43E-08
Health/Safety	21.80	7.84	19.11	7.02	2.44	105.90	1.65E-02
Recreational	23.01	9.40	25.56	10.07	-1.69	90.54	9.54E-02
Social	32.42	6.44	29.77	5.62	2.97	108.63	3.67E-03

Risk Domain	School μ	School σ	USB μ	USB σ	t	df	p
Ethical	11.97	4.15	12.82	4.96	-0.85	66.05	4.00E-01
Financial	13.90	6.15	15.32	5.22	-1.06	48.97	2.93E-01
Health/Safety	16.14	6.28	19.11	7.02	-1.99	62.31	5.11E-02
Recreational	18.21	6.44	25.56	10.07	-4.11	79.49	9.70E-05
Social	27.34	6.61	29.77	5.62	-1.69	49.07	9.71E-02

WHO DID IT?

Category	Flash Drive		University	p
Age[12]				
18-20	20/55	(36%)	38%	0.90
21-29	32/55	(58%)	55%	0.75
30-39	1/55	(2%)	6%	0.37*
40+	2/55	(4%)	1%	0.12*
Affiliation				
Undergraduate	41/62	(66%)	59%	0.34
Graduate	13/62	(21%)	20%	0.99
Staff	7/62	(11%)	15%	0.50
Faculty	0/62	(0%)	5%	0.08*
Prefer not to answer	1/62	2%	–	–

TABLE VII: **Demographics**—We collect demographic information about participants who plugged in the flash drives and find that they do not significantly differ from the University population.
* Comparison performed using Fisher's Exact Test instead of the test of equal proportions.

Representative of the university setting

DID THEY KNOW WHAT THEY WERE DOING?

Security Behavior Intentions Scale (SeBIS)

Original study: Mechanical Turks. Not representative of UIUC

Question[15]	Egelman and Peer μ	Egelman and Peer σ	USB μ	USB σ	t	df	p
I set my computer screen to automatically lock if I don't use it for a prolonged period of time.	3.20	1.559	3.95	1.419	-3.790	75.510	2.98E-04
I use a password/passcode to unlock my laptop or tablet.	3.78	1.525	4.19	1.420	-2.060	74.700	4.26E-02
I manually lock my computer screen when I step away from it.	2.63	1.343	3.32	1.514	-3.360	69.210	1.27E-03
I use a PIN or passcode to unlock my mobile phone.	3.21	1.733	3.75	1.677	-2.310	73.400	2.36E-02
I do not change my passwords, unless I have to[r].	2.65	1.091	1.88	1.001	5.520	75.210	4.59E-07
I use different passwords for different accounts that I have.	3.75	1.037	3.19	1.152	3.590	69.550	6.11E-04
I do not include special characters in my password if it's not required[r].	3.30	1.292	2.85	1.472	2.260	68.960	2.69E-02
When someone sends me a link, I open it without first verifying where it goes[r].	4.01	1.014	2.95	1.209	6.470	67.970	1.24E-08
I submit information to websites without first verifying that it will be sent securely (e.g., SSL, "https://", a lock icon)[r].	3.69	1.102	3.31	1.149	2.440	71.190	1.70E-02
When browsing websites, I mouseover links to see where they go, before clicking them.	3.69	1.027	3.25	1.359	2.380	66.040	2.00E-02
If I discover a security problem, I continue what I was doing because I assume someone else will fix it[r].	4.08	0.976	3.71	1.115	2.430	68.900	1.78E-02
When I'm prompted about a software update, I install it right away.	3.07	1.035	2.81	1.008	1.840	73.190	6.94E-02
I try to make sure that the programs I use are up-to-date.	3.78	0.890	3.53	0.935	1.990	70.970	5.07E-02

Question	School μ	School σ	USB μ	USB σ	t	df	p
I set my computer screen to automatically lock if I don't use it for a prolonged period of time.	3.36	1.471	3.95	1.419	1.770	51.450	8.21E-02
When I'm prompted about a software update, I install it right away.	3.36	1.026	2.81	1.008	-2.320	52.290	2.42E-02

TABLE IX: **SeBIS Results**—We compare items with different ($p < 0.1$) responses to items in the SeBIS in both Egelman and Peer's study [12] and the USB experiment and between the school survey and the USB experiment. College students appear to have different security knowledge profiles than a general population.

TODAY'S PAPERS

USERS ARE NOT THE ENEMY

Anne Adams & Martina Angela Sasse
Department of Computer Science
University College London

Many system security departments treat users as a security risk to be controlled. The general consensus is that most users are careless and unmotivated when it comes to system security. In a recent study, we found that users may indeed compromise computer security mechanisms, such as password authentication, both knowing and unknowingly. A closer analysis, however, revealed that such behavior is often caused by the way in which security mechanisms are implemented, and users' lack of knowledge. We argue that to change this state of affairs, security departments need to communicate more with users, and adopt a user-centered design approach.

Introduction

Confidentiality is an important aspect of computer security. It is dependent on authentication mechanisms, such as passwords, to safeguard access to information [9]. Traditionally, authentication procedures are divided into two stages: *identification* (User ID), to identify the user and *authentication*, to verify that the user is the legitimate owner of the ID. It is the latter stage that requires a secret password. To date, research on password security has focused on designing technical mechanisms to protect access to systems; the usability of these mechanisms has rarely been investigated. Hitchings [8] and Davis & Price [4] argue that this narrow perspective has produced security mechanisms which are, in practice, less effective than they are generally assumed to be. Since security mechanisms are designed, implemented, applied and breached by people, human factors should be considered in their design. It seems that currently, hackers pay more attention to the human link in the security chain than security designers do, e.g. by using *social engineering* to obtain passwords.

The key element in password security is the *crackability* of a password combination. Davies & Ganesan [3] argue that an adversary's ability to crack passwords is larger than usually believed. System-generated passwords are essentially the optimal security approach; however, user-generated password are potentially more memorable and thus less likely to be disclosed (e.g. because users have write them down). The US Federal Information Processing Standards [5] suggest several criteria for assuring different levels of password security. *Password composition*, for example, relates the size of a character set from which a password has been chosen to its level of security. An alpha-numeric password is therefore more secure than one composed of letters alone. Short *password lifetime* - i.e. changing passwords frequently - is suggested as reducing the risk associated with undetected compromised passwords. Finally, *password ownership*, in particular individual ownership, is suggest to:

- increase individual accountability;
- reduce illicit usage;
- allow for an establishment of system usage audit trails;
- reduce frequent password changes due to group membership fluctuations.

There is evidence that many password users do not comply with these suggested rules. DeAlvare [1] found that once a password is chosen, a user is unlikely to change it until it has been shown to be compromised. Users were also found to construct passwords that contained as few characters as possible [2]. These observations cannot be disputed, but the

Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0

Alma Whitten
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
alma@cs.cmu.edu

J. D. Tygar[1]
EECS and SIMS
University of California
Berkeley, CA 94720
tygar@cs.berkeley.edu

Abstract

User errors cause or contribute to most computer security failures, yet user interfaces for security still tend to be clumsy, confusing, or near-nonexistent. Is this simply due to a failure to apply standard user interface design techniques to security? We argue that, on the contrary, effective security requires a different usability standard, and that it will not be achieved through the user interface design techniques appropriate to other types of consumer software.

To test this hypothesis, we performed a case study of a security program which does have a good user interface by general standards: PGP 5.0. Our case study used a cognitive walkthrough analysis together with a laboratory user test to evaluate whether PGP 5.0 can be successfully used by cryptography novices to achieve effective electronic mail security. The analysis found a number of user interface design flaws that may contribute to security failures, and the user test demonstrated that when our test participants were given 90 minutes in which to sign and encrypt a message using PGP 5.0, the majority of them were unable to do so successfully.

We conclude that PGP 5.0 is not usable enough to provide effective security for most computer users, despite its attractive graphical user interface, supporting our hypothesis that user interface design for effective security remains an open problem. We close with a brief description of our continuing work on the development and application of user interface design principles and techniques for security.

1 Introduction

Security mechanisms are only effective when used correctly. Strong cryptography, provably correct protocols, and bug-free code will not provide security if the people who use the software forget to click on the encrypt button when they need privacy, give up on a communication protocol because they are too confused about which cryptographic keys they need to use, or accidentally configure their access control mechanisms to make their private data world-readable. Problems such as these are already quite serious: at least one researcher [2] has claimed that configuration errors are the probable cause of more than 90% of all computer security failures. Since average citizens are now increasingly encouraged to make use of networked computers for private transactions, the need to make security manageable for even untrained users has become critical [4, 9].

This is inescapably a user interface design problem. Legal remedies, increased automation, and user training provide only limited solutions. Individual users may not have the resources to pursue an attacker legally, and may not even realize that an attack took place. Automation may work for securing a communications channel, but not for setting access control policy when a user wants to share some files and not others. Employees can be required to attend training sessions, but home computer users cannot.

Why, then, is there such a lack of good user interface design for security? Are existing general user interface design principles adequate for security? To answer these questions, we must first understand what kind of usability security requires in order to be

[1] Also at Computer Science Department, Carnegie Mellon University (on leave).