Usable Security

What Is It and Why Do We Need It?

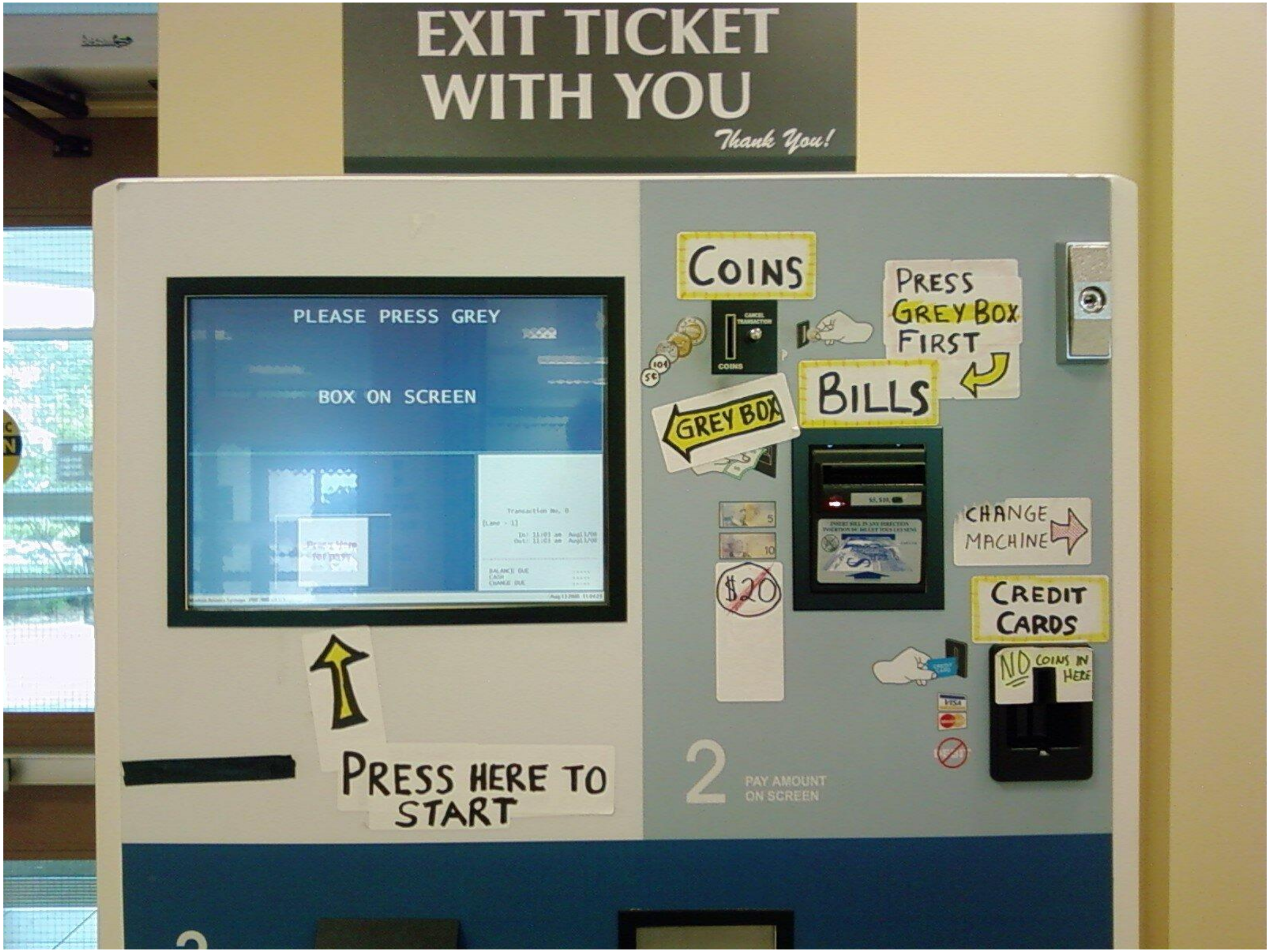
Karoline Busse

Usable Security and Privacy Research Group, Universität Bonn

Overview

Usability + Security

Why We Need Usability



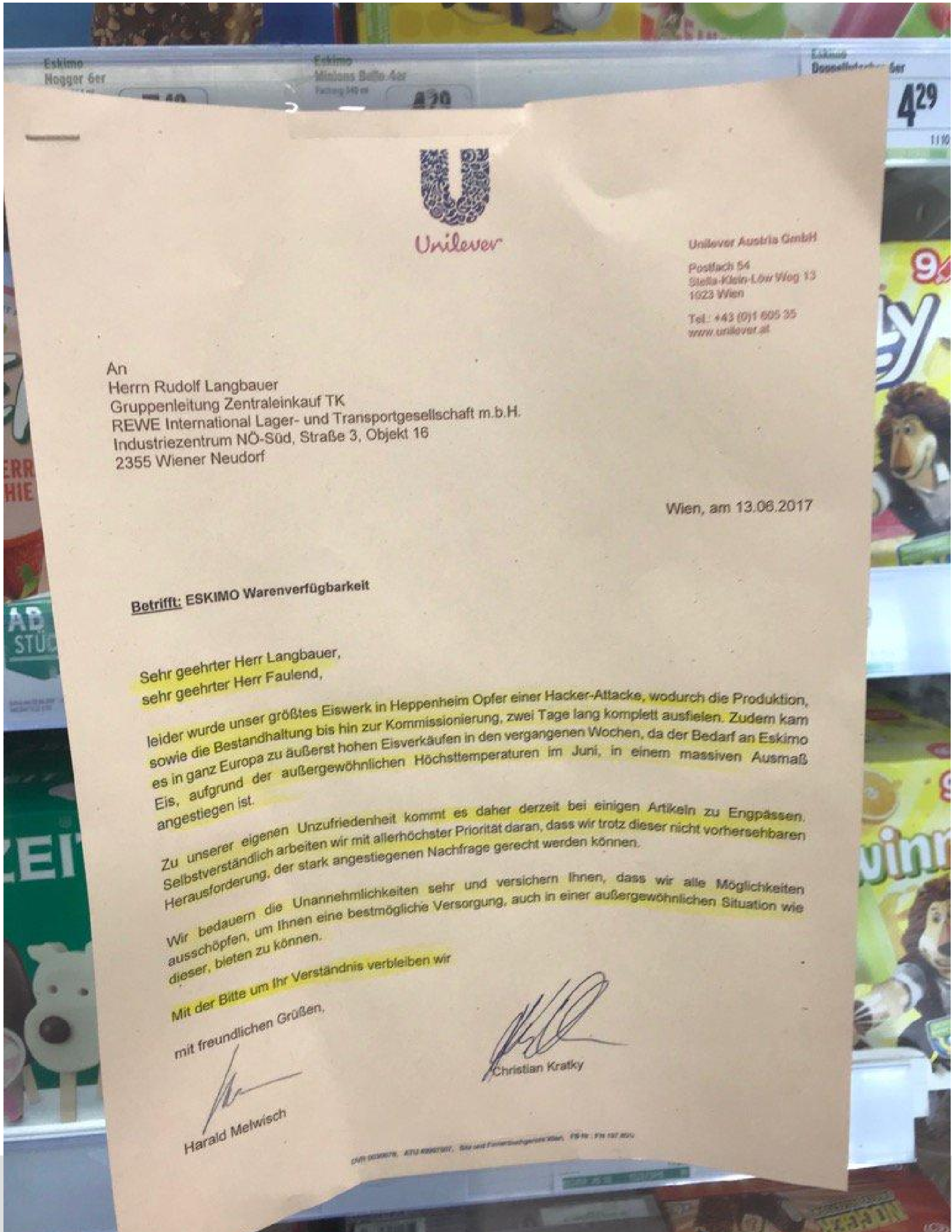
CC-BY-ND Rick Dolishny

Why We Need Security

„Leider wurde unser größtes Eiswerk in Hoppenheim Opfer einer Hacker-Attacke, wodurch die Produktion […] zwei Tage lang komplett ausfielen.“

[…]

Zu unserer eigenen Unzufriedenheit kommt es daher derzeit bei einigen Artikeln zu Engpässen.“


Eskimo
Nogger 6er
Eskimo
4,29

Unilever

Unilever Austria GmbH

Postfach 54
Stella-Klein-Löw Weg 13
1023 Wien

Tel.: +43 (0)1 605 35
www.unilever.at

An
Herrn Rudolf Langbauer
Gruppenleitung Zentraleinkauf TK
REWE International Lager- und Transportgesellschaft m.b.H.
Industriezentrum NÖ-Süd, Straße 3, Objekt 16
2355 Wiener Neudorf

Wien, am 13.06.2017

Betrifft: **ESKIMO Warenverfügbarkeit**

Sehr geehrter Herr Langbauer,
sehr geehrter Herr Faulend,

leider wurde unser größtes Eiswerk in Heppenheim Opfer einer Hacker-Attacke, wodurch die Produktion, sowie die Bestandhaltung bis hin zur Kommissionierung, zwei Tage lang komplett ausfielen. Zudem kam es in ganz Europa zu äußerst hohen Eisverkäufen in den vergangenen Wochen, da der Bedarf an Eskimo Eis, aufgrund der außergewöhnlichen Höchsttemperaturen im Juni, in einem massiven Ausmaß angestiegen ist.

Zu unserer eigenen Unzufriedenheit kommt es daher derzeit bei einigen Artikeln zu Engpässen. Selbstverständlich arbeiten wir mit allerhöchster Priorität daran, dass wir trotz dieser nicht vorhersehbaren Herausforderung, der stark angestiegenen Nachfrage gerecht werden können.

Wir bedauern die Unannehmlichkeiten sehr und versichern Ihnen, dass wir alle Möglichkeiten ausschöpfen, um Ihnen eine bestmögliche Versorgung, auch in einer außergewöhnlichen Situation wie dieser, bieten zu können.

Mit der Bitte um Ihr Verständnis verbleiben wir

mit freundlichen Grüßen,

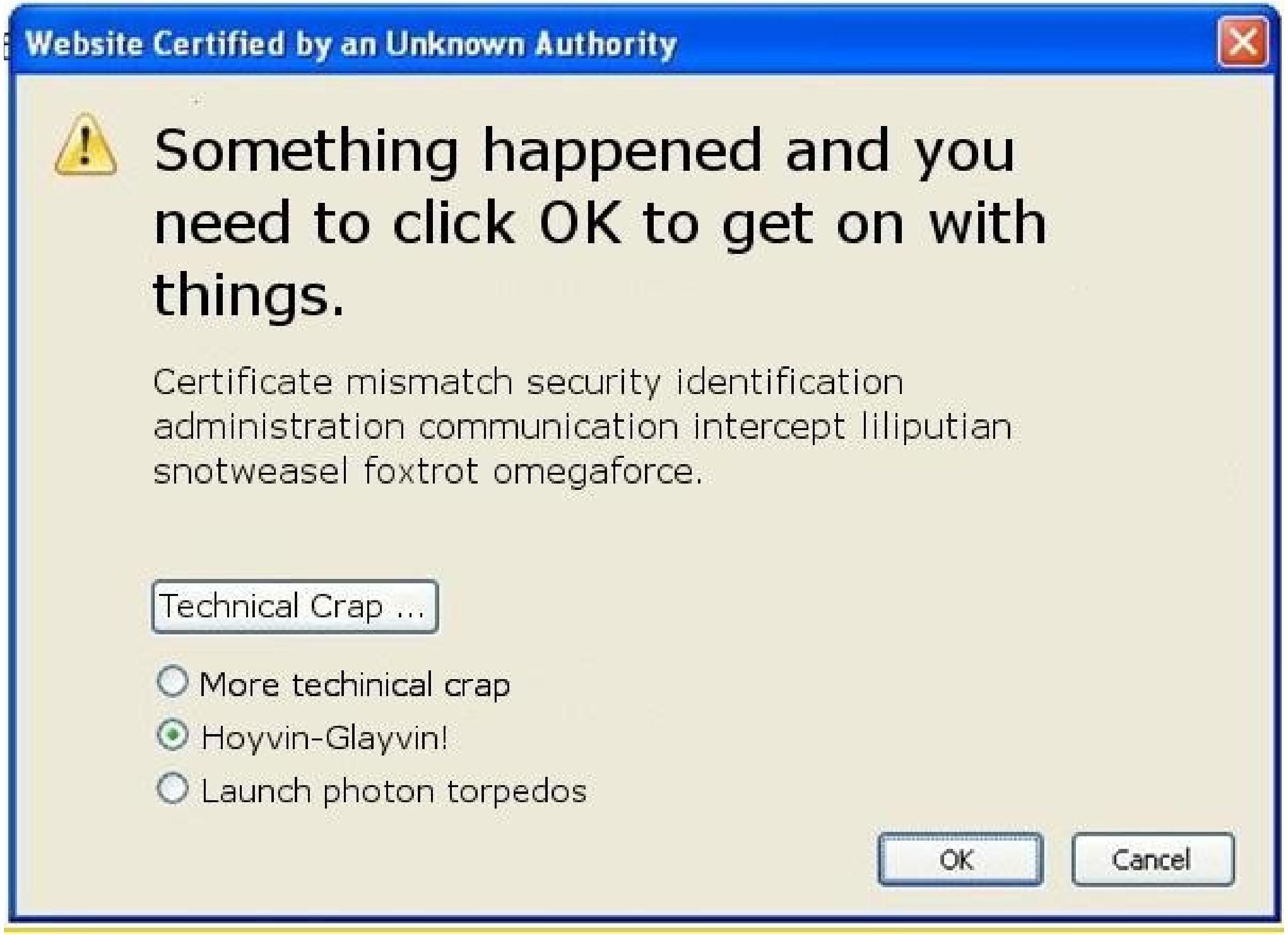
Harald Melwisch

Christian Kratky

DVR [illegible], ATU [illegible], Sitz und Firmenbuchgericht Wien, FB-Nr.: FN 137 [illegible]

Pic: @skoops/Twitter

Why Do We Need Usable Security?



Website Certified by an Unknown Authority

Something happened and you need to click OK to get on with things.

Certificate mismatch security identification administration communication intercept liliputian snotweasel foxtrot omegaforce.

○ More techinical crap

● Hoyvin-Glayvin!

○ Launch photon torpedos

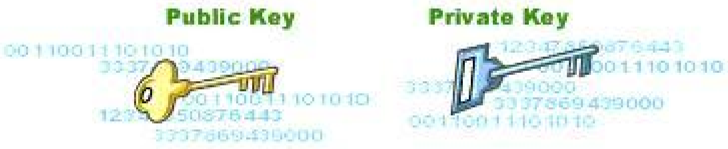
Adapted from Jonathan Nightingale

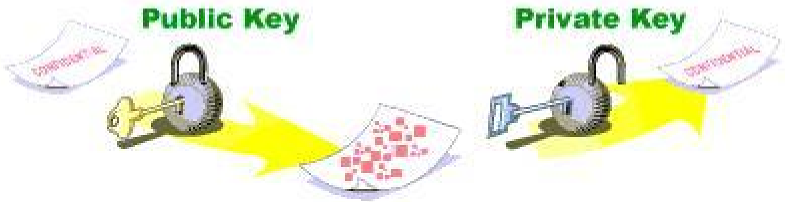
Because Security is Hard. We want to make it easy!

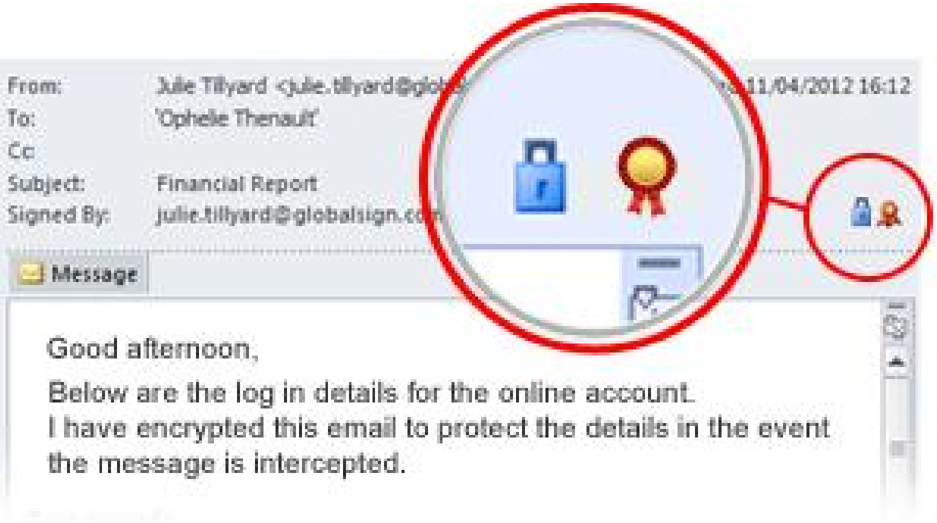
Usable Security Origins

- Three seminal papers are seen as the origin of Usable Security and Privacy research:
    - 1996 Zurko and Simon’s: “User-Centered Security”
    - 1999 Adams and Sasse’s: “Users Are Not the Enemy”
    - 1999 Whitten and Tygar’s “Why Johnny Can’t Encrypt: A Usability Evaluation of PGP 5.0”
        - USENIX Security Test of Time Award 2015
- All argued that users should not be seen as the problem to be dealt with,
    - but that security experts need to communicate more with users, and adopt user-centered design approaches.

PGP: The Classic







From: Julie Tillyard <julie.tillyard@glob… 11/04/2012 16:12

To: 'Ophelie Thenault'

Cc:

Subject: Financial Report

Signed By: julie.tillyard@globalsign.co…

Message

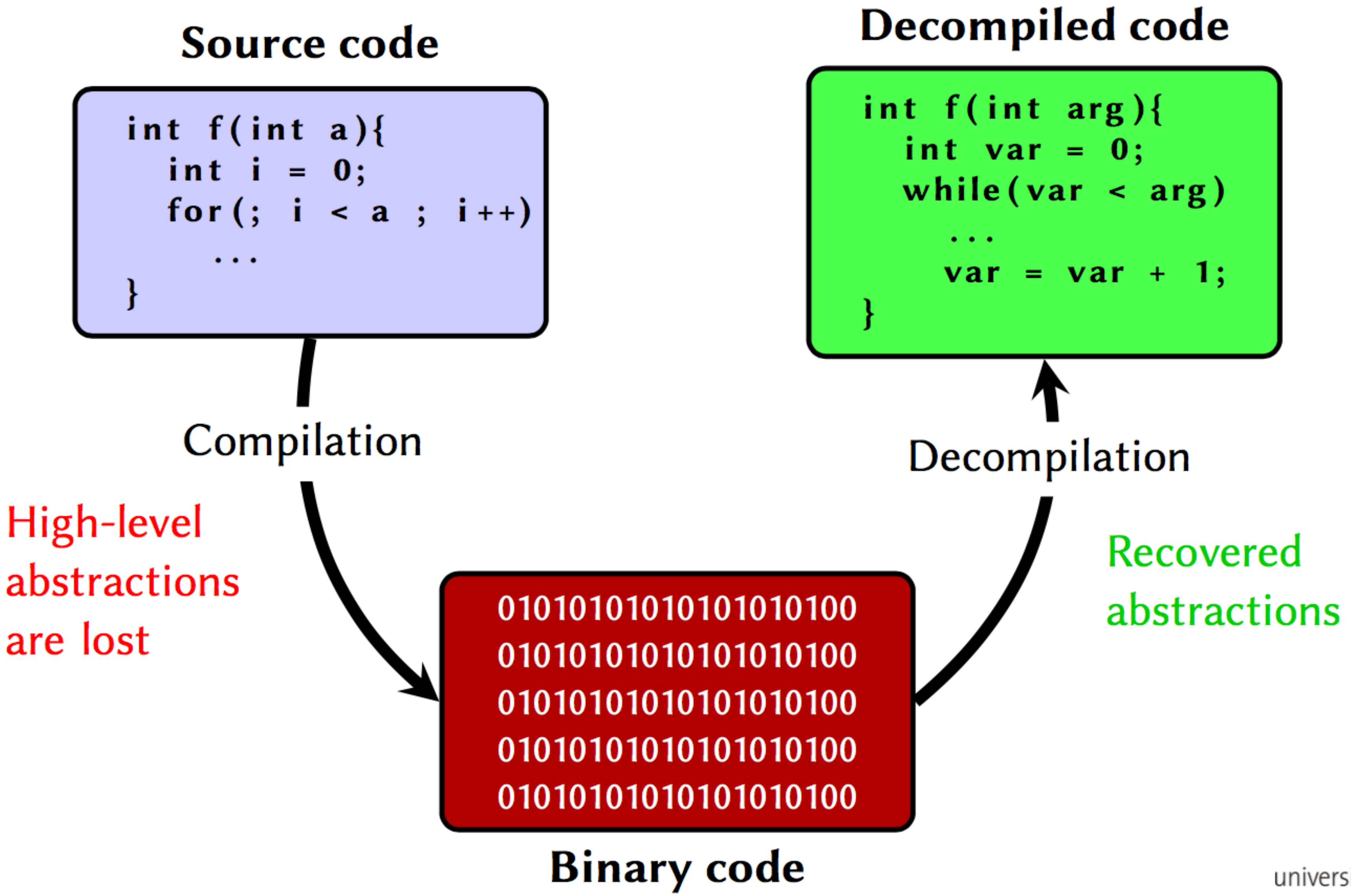
Good afternoon,

Below are the log in details for the online account.
I have encrypted this email to protect the details in the event the message is intercepted.

What We Do

Usable Security for Professionals

Facilitating Malware Analysis



Source code

```
int f(int a){
  int i = 0;
  for(; i < a ; i++)
     ...
}
```

Compilation

High-level abstractions are lost

```
01010101010101010100
01010101010101010100
01010101010101010100
01010101010101010100
01010101010101010100
```

Binary code

Decompilation

Recovered abstractions

Decompiled code

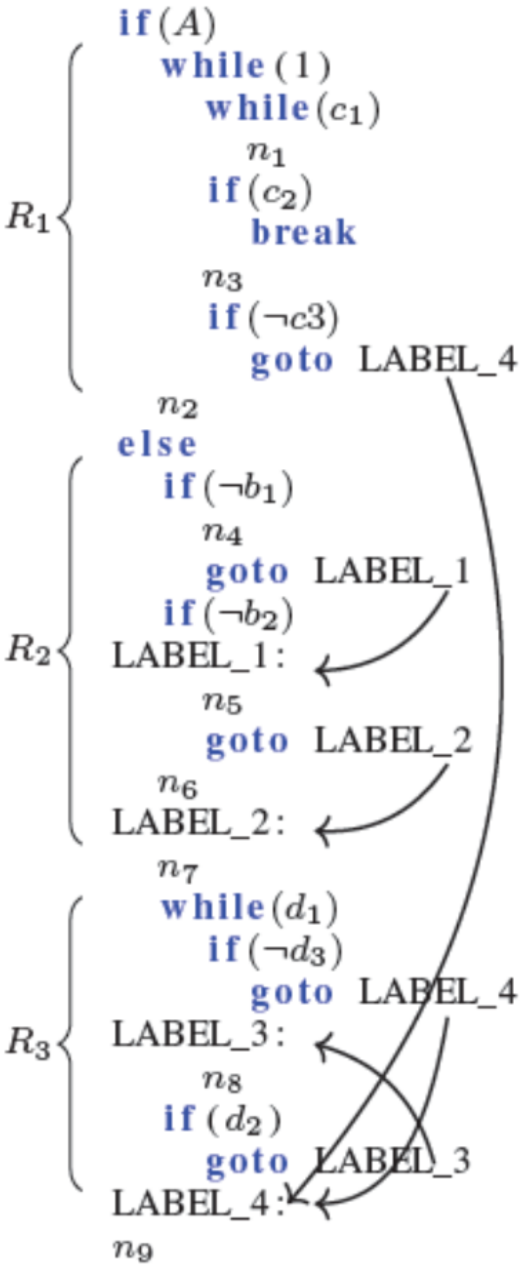
```
int f(int arg){
  int var = 0;
  while(var < arg)
    ...
    var = var + 1;
}
```

universitä

Facilitating Malware Analysis

Decompiling a P2P Zeus sample with Hex-Rays

- 1,571 goto for 49,514 LoC
- 1 goto for each 32 LoC



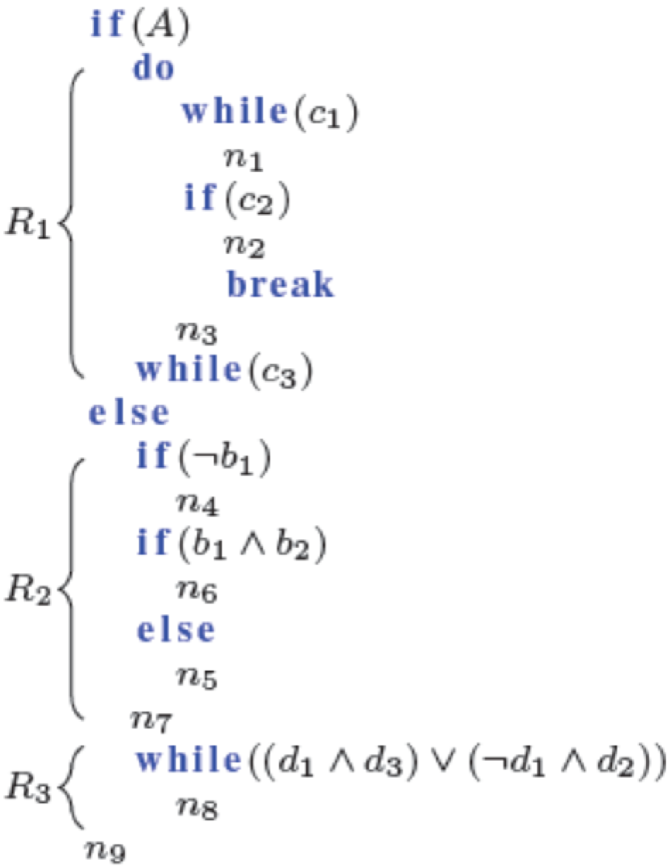
```
if (A)
    while (1)
        while (c1)
            n1
        if (c2)
            break
        n3
        if (¬c3)
            goto LABEL_4
    n2
else
    if (¬b1)
        n4
        goto LABEL_1
    if (¬b2)
LABEL_1:
        n5
        goto LABEL_2
    n6
LABEL_2:
    n7
    while (d1)
        if (¬d3)
            goto LABEL_4
LABEL_3:
        n8
    if (d2)
        goto LABEL_3
LABEL_4:
n9
```

DREAM Decompiler

- DREAM Decompiler
  - No more gotos!
  - Most compact code

NDSS’15 Distinguished Paper: „No More Gotos: Decompilation Using Pattern-Independent Control-Flow Structuring and Semantic-Preserving Transformations

- DREAM++ Decompiler
  - Additional usability improvements
  - Conducted quantitative user study

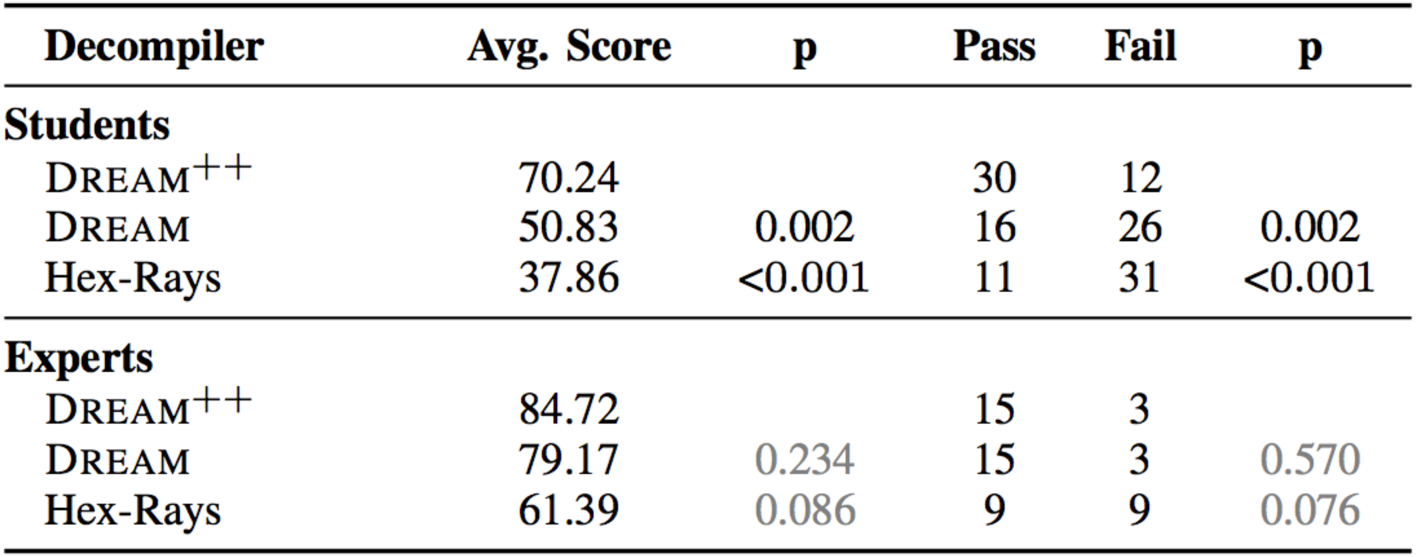


```
if ($A$)
    do
        while ($c_1$)
            $n_1$
        if ($c_2$)
            $n_2$
            break
        $n_3$
    while ($c_3$)
else
    if ($\neg b_1$)
        $n_4$
    if ($b_1 \wedge b_2$)
        $n_6$
    else
        $n_5$
    $n_7$
    while (($d_1 \wedge d_3$) $\vee$ ($\neg d_1 \wedge d_2$))
        $n_8$
$n_9$
```

DREAM++ Evaluation

IEEE S&P ‘16: „Helping Johnny to Analyse Malware: A Usability-Optimized Decompiler and Malware Analysis User Study“

- 3 decompilers (within-subjects)
  - Hex-Rays
  - DREAM
  - DREAM++
- 2 levels of experience (between-subject)
  - Students and Professionals
- 2 groups of malware analysis tasks (split-plot)
  - 3 medium and 3 hard tasks (within-subjects)



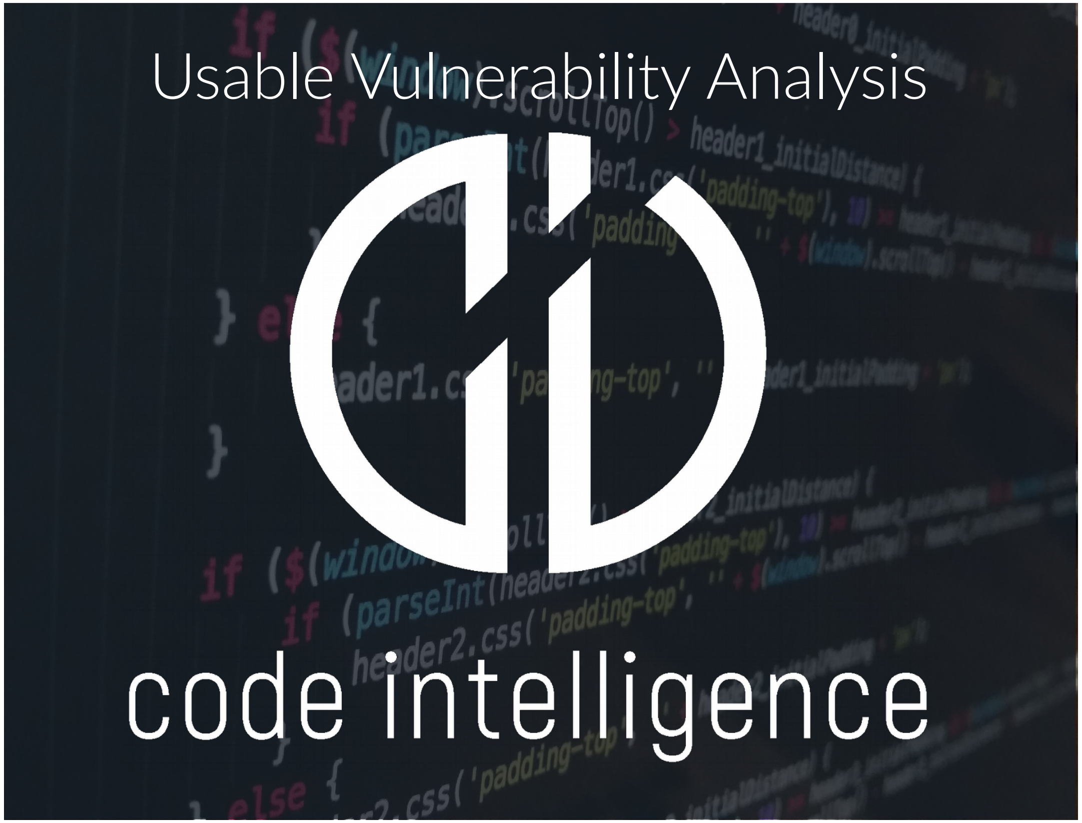
Decompiler	Avg. Score	p	Pass	Fail	p
Students					
DREAM++	70.24		30	12	
DREAM	50.83	0.002	16	26	0.002
Hex-Rays	37.86	<0.001	11	31	<0.001
Experts					
DREAM++	84.72		15	3	
DREAM	79.17	0.234	15	3	0.570
Hex-Rays	61.39	0.086	9	9	0.076

Follow-Up Research and Startup

- Follow-Up: Function Recognition in Binaries
  - Cooperation with Politecnico di Milano
  - To be published 2018

- Startup:

  Code Intelligence



ERC Research Grant:
Frontiers of Usable Security

ERC Grant: USec Frontiers

- Password storage is hard
  - See latest password breaches (Yahoo et al.)
- Where do developers struggle?
  - Researching password storage APIs in Java

Why is password storage so hard?

- 2 frameworks (between-subjects)
  - JSF (manual implementation)
  - Spring (opt-in)
- 2 levels of priming (between-subject)
  - With or without security emphasis
- Pre-screening survey and debriefing interview

Findings

- Security knowledge does not guarantee secure software
- More usable APIs are not enough
  - Secure password storage needs to be enforced
- Explicitly requesting security is necessary
- Continious Learning: Many implemented outdated mechanics
- Conflicting advice on secure storage makes it hard

ACM CCS 2017: „Why Do Developers Get Password Storage Wrong?
A Qualitative Usability Study“

Perception of
Security and Privacy

Security Perception

- Different people have different conceptions of security and privacy
  - Example: Threat modeling
  - This shapes security decisions and habits

- Products are often designed with a Western (US/EU) audience in mind
  - How does this influence adaption in other cultural contexts (e.g. Asia, Middle East)

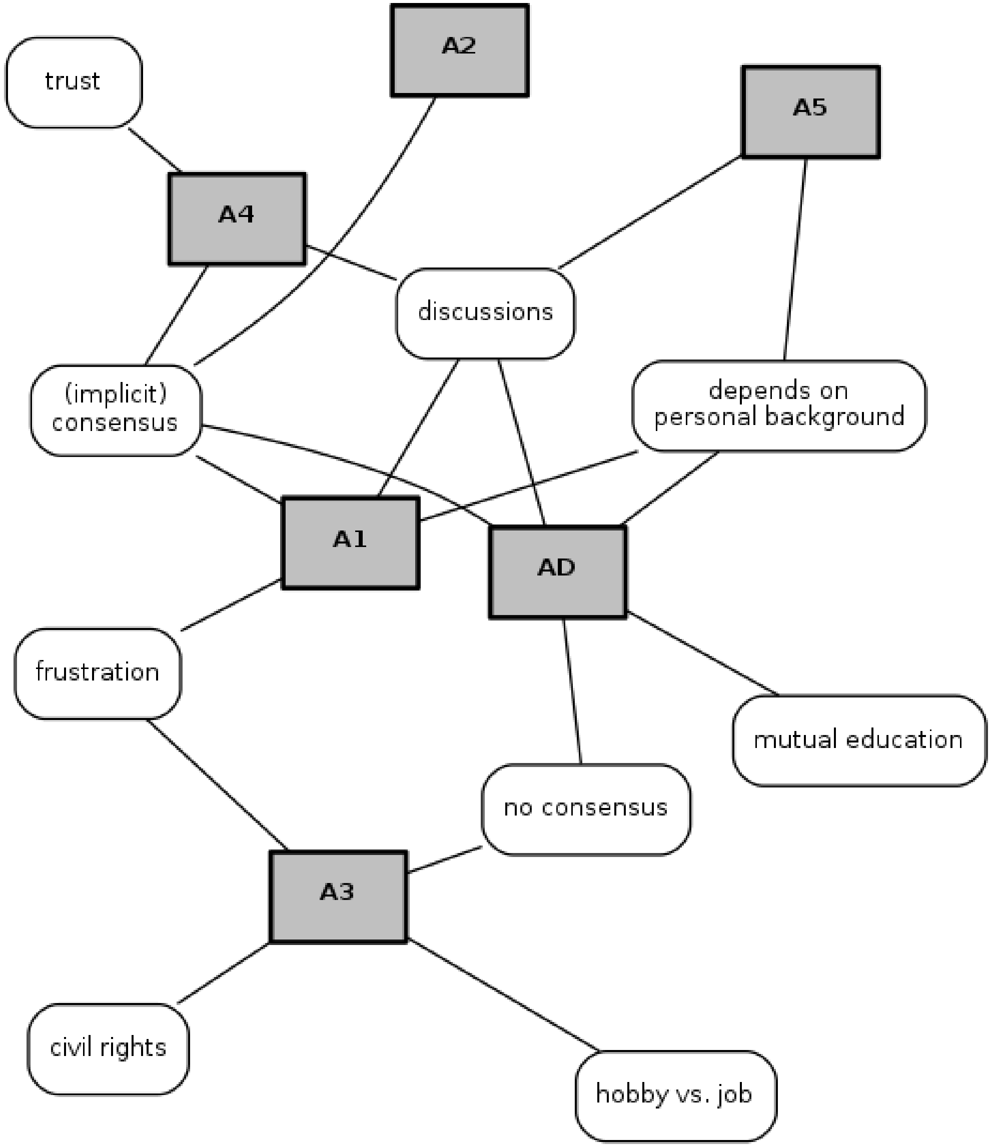
- In the work context, seucirty narratives can influence employee happiness and internal power struggles

Security Narratives

- Interview study within a small consulting company
  - 5 employees, 1 CEO

- Surprising findings:
  - Uncertainty is not necessarily a bad thing
  - Fruitful discussions and mutual education

EuroUSEC 2017 WIP: „Security Narratives: Can (Language) Insecurities be Beneficial for Security Departments?“



Methods in Usable Security Research

UseME

- Brand new sibling research group since autumn 2017
  - Head: Emanuel von Zezschwitz
  - Heavily HCI influenced
- Methodological research:
  - Lab studies vs. Field studies
  - Experts vs. End users
- Mobile HCI, Privacy and Security

Questions?