Bad UX is Bad Security
Adventures in Qubes OS UX Design
by Marta Marczykowska-Górecka (marmarta)
Why UX matters for security?

• Theoretical security vs practical security
• It doesn’t matter whose fault it is – the harm is done
• User errors are a real and important vector of attack
• Treat users seriously, not like children who need to behave better
The Human Factor

- Humans make mistakes
- Humans might have other priorities than using the software perfectly
- Human brains are not optimized for the kind of tasks we want them to do with computers
Shortcuts

• If people keep using a shortcut, there’s a need that has to be fulfilled

• The goal of doing a thing with a computer is rarely “do security” – security is a desired trait, but not goal in and of itself
Mistakes

- People make mistakes and will always make mistakes
- Even the smartest person in the room can be in a hurry

Not great. Not terrible.
The Problem of Attention

• Inattentitional blindness: we only notice the things we care about

• Cocktail party phenomenon – this is generally a good thing for humans, but annoying for developers and designers
The Invisible Error Message

Document Could Not Be Locked

The lock file could not be created for exclusive access by LibreOffice, due to missing permission to create a lock file on that file location or lack of free disk space.

Open Read-Only  Cancel
The Invisible Error Message

NOPE

blah blah blah words words words

DO MY THING  Cancel
What is Qubes OS?

• A meta-operating system

• Security through compartmentalization: instead of running everything together, the system is partitioned into isolated virtual machines called qubes

• Basically: bunch of virtual machines in a fancy comfortable trenchcoat
Qubes OS: Simplified Introduction

- **sys-usb**: holds all USB devices
- **sys-net**: deals with networking devices
- **sys-firewall**: handles firewall rules
- **social-media**: browser & image editing
- **work**: various work-related programs
- **email**: browser & mail
- **disposable**: browser, this qube vanishes after use

**Fedora template**
serves as a base for other qubes

**Debian template**
serves as a base for other qubes

**System qubes**

**User qubes**
Case study: Qubes OS copy and paste

Normal Linux/Windows copy-paste

1. select text
2. Ctrl + C
3. text is in clipboard
4. Ctrl + V
5. the text is in a new place
Case study: Qubes OS copy and paste

Normal Linux/Windows copy-paste

Qubes OS copy-paste
Case study: copy and paste

- Not perfect: people do get used to the extra step and it becomes nearly as automatic as the default
- But this workflow protects against things like clipboard stealing
- Further security is of course still necessary, thus: policy
Case study: copy and paste

Clipboard Shortcuts

Qubes OS features a secure "inter-qube" or "global" clipboard that allows you to copy and paste between qubes while preventing any qube other than your selected target from stealing content from the clipboard. Without such a system, any content copied to the global clipboard, such as a password, would instantly be exposed to every other running qube, including qubes you don't trust. By giving you precise control over exactly which qube receives inter-qube clipboard content, then immediately wiping the inter-qube clipboard afterward, Qubes OS protects the confidentiality of the text being copied.

Inter-qube copy and paste actions are performed via special keyboard shortcuts, as specified below. These keyboard shortcuts are always intercepted by dom0 (so that rogue qubes can't perform global copy/paste actions on their own).

*Note*: Changes below require a qube restart to take effect.

Copy keyboard shortcut: default (Ctrl+Shift+C)  
Paste keyboard shortcut: default (Ctrl+Shift+V)

Clipboard Policy

Prevent accidental errors when using the inter-qube (global) clipboard.

- **Default policy.** Allow any qube to copy/paste into any other qube, except dom0.
## Case study: copy and paste

**Custom policy.** Specify which qubes are allowed to copy/paste into which other qubes.

<table>
<thead>
<tr>
<th>Origin qube</th>
<th>Permission</th>
<th>Destination qube</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL QUBES</td>
<td>will</td>
<td>always</td>
</tr>
<tr>
<td></td>
<td></td>
<td>be allowed to paste into clipboard of ALL QUBES</td>
</tr>
</tbody>
</table>

With the following exceptions:

- **vault**
  - will
  - never
  - be allowed to paste into clipboard of ALL QUBES

- **TYPE: ADMINVM**
  - will
  - always
  - be allowed to paste into clipboard of ALL QUBES

+ADD new
Case study: devices

• The problem: devices are evil
• And when they are not evil, they definitely can do too much (see: microphone and camera)
• Qubes OS isolates devices in their own qube and allows the user to connect them as needed to other qubes
Case study: devices: old view
Case study: devices
Case study: devices

- Storage Devices
  - Kingston USB
  - Android Phone
  - Media Devices
    - Microphone
    - Generic_Chicony_Camera_USB_Camera_Device
  - Other Devices
    - Logitech Mouse
    - Unknown

- Device settings

- Kingston USB
  - Detach from personal
  - Detach and attach to favorite qube: signal
  - Detach and attach to other qube: anon-whonix, sys-firewall, sys-net
  - Detach and attach to new disposable qube (fedora-32-dvm)
How to Design for Security

• Design for error, not just for success
• Secure should be easy, insecure hard
• Design for actual humans, not for perfect people
• Cutting corners will happen: plan for it