

Implementing digital examinations in free software systems

Juha Erkkilä

puavo.org

- I have been a free software hobbyist since 1999
- I love BSD Unix but worked professionally with Linux since 2008
- I am one of the main developers of the Puavo-project, a free software project by Opinsys



- Puavo consists of an operating system (Puavo OS) that is based on Debian GNU/Linux, and software for managing a large number of computers
- developed for schools and used in Finland and Germany
- licensed under GNU General Public License version 2 +
- company-developed, hopefully there could be a community behind it some day
- <https://github.com/puavo-org/>

Digital examinations in Finland

Finnish Matriculation Examination

- the final examinations for Finnish upper secondary schools have been digital for some years now (fully from 2019 onwards)
- the Matriculation Examination Board is responsible for administering the examinations
- they decide on the technologies used in the examinations, these choices affect all upper secondary school students

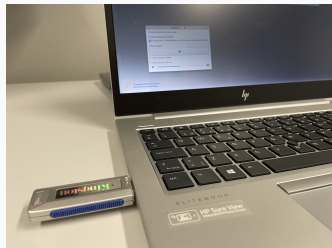
Abitti digital exam environment

- commonly the same system that is used for final exams is also used for course examinations during normal school year
- laptops owned by students are booted from an USB-drive to a separate exam operating system, which is based on Debian
- the exam OS connects to a server that manages the exam sessions

- the security requirements in the final examinations are quite strict
- mice with programmable macros are banned (could be used to output exam answers)
- possible information leaks into examinations include changing network SSIDs that could be read from networking applets
- phones, smart watches, smart rings, wireless devices are forbidden

Why use Linux USB-drive for exams?

- as the USB-drive is distributed right before the exam:
 - the OS security on the main hard drive is irrelevant
 - the OS security in the exam environment is fully controlled
- meaning that the same laptops used by students (possibly owned by students) can be used for the exam



Why *not* use Linux USB-drive for exams?

- laptops need to support Linux fairly well
 - to help buyers choose exam-compatible laptops, these are marked in computer stores as such
 - the sticker on the right essentially means "exam-compatible"
- managing USB-drives for course examinations during school year is a hassle (they need to be updated periodically)
- USB-drives are less reliable than hard drives



Linux on the exam client laptops: problems

- schools want to choose computer hardware freely dismissing Linux-compatibility
- Macbooks do not work (is a different architecture to support)
- Chromebooks have OS verification features that prevent or make it hard to boot alternative systems
- many recent Windows-laptops have issues

Matriculation Examination Board: Linux-support in the future

- on the green: computers supported by Linux (the exam OS)
- on the red: computers *not* supported by Linux



The death of Linux-based USB drives

- a decision made in 2023 by the Matriculation Examination Board: in the future, Linux-based USB-drives will not be used for the student exam environment
- will be replaced by a cross-platform "examination application"
 - supporting Windows
 - supporting Mac OS
 - supporting Chromebooks
- a decision not opposed by many (non-techies)

Examination application

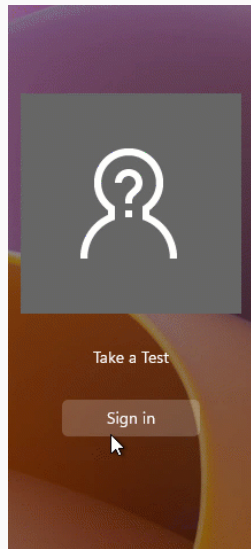
What is an "examination application"?

- an application for taking an exam
- it should:
 - lock the student out of their normal programs and files during the exam session so they can not cheat
 - monitor student actions on their computer during the exam session so if they cheat, they will get caught
 - allow normal computer operation only after the student has finished the exam
- now how to implement this in an "app"?

Operating systems must provide such features? (Windows)

from <https://learn.microsoft.com/en-us/education/windows/take-a-test-app-technical>:

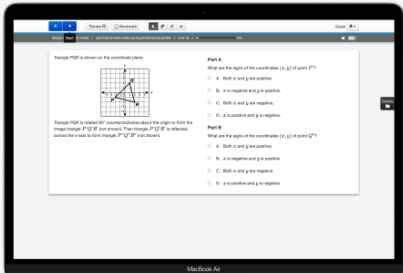
- *"Take a Test is an application that locks down a device and displays an online assessment web page."*
- *"When the assessment page initiates lock-down, the student's desktop is locked and the app executes above the Windows lock screen. This provides a sandbox that ensures the student can only interact with the Take a Test app."*



Operating systems must provide such features? (Mac OS)

from <https://education-static.apple.com/leadership/k12-assessment-overview.pdf>:

- *"iPad and Mac are approved devices for administering secure exams and standardized assessments in every state."*
- *"In assessment mode, Mac will automatically lock into a single app."*



Triangle PQR is shown on the coordinate plane.

Triangle PQR is rotated 90° counterclockwise about the origin to form the image triangle P'Q'R'. What are the signs of the coordinates (x, y) of point P'?

Part A

What are the signs of the coordinates (x, y) of point P'?

A. Both x and y are positive.

B. x is negative and y is positive.

C. Both x and y are negative.

D. x is positive and y is negative.

Part B

What are the signs of the coordinates (x, y) of point Q'?

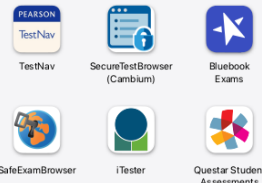
A. Both x and y are positive.

B. x is negative and y is positive.

C. Both x and y are negative.

D. x is positive and y is negative.

Testing apps compatible with assessment mode for Mac



TestNav

SecureTestBrowser (Cambium)

Bluebook Exams

SafeExamBrowser

iTester

Questar Student Assessments

Operating systems must provide such features? (Chrome OS)

from

<https://support.google.com/chrome/a/answer/3273084?hl=en>

- *"When set up properly, Chromebooks meet K–12 education testing standards and are a secure platform for student assessments. You can disable students' access to browse the web during an exam, external storage, screenshots, and the ability to print."*
- Exams can be setup through "kiosk mode"

Operating systems must provide such features? (Linux)

- What about examination mode under Linux?

Wait... what are we actually doing here?

- during the exam session, the computer should be controlled by those who operate the exam, and not the student
- the same computer is to be used normally outside the exam

- the USB-drive scheme solves the problem of two masters: computer normally controlled by user, but on the exam by the exam overlords
 - there are some safety implications to user on that setting, but mostly acceptable
- why not use two sets of computers then, a special set for examinations?
 - the problem is economics and convenience: it is nice for students to take exams on the same computers they may use for homework

Free software is about freedom... to cage users?

- the point of free software (and Linux) is to put users in control
- if users can manage the OS, they can subvert the examination application
- ... it logically follows that systems must be managed by schools or some other organisation and users must not have full control

Implementing system security and examination mode

- when I was young, it was commonly accepted wisdom that having physical access to a computer implied full system access
- that does not have to be the case anymore
- for the exam mode to be secure against students:
 - BIOS must be secured
 - boot process must be secured
 - system integrity must be ensured
 - deny root-access

System security - BIOS and SecureBoot

- keep BIOS admin password secret: students should use computers where they do *not* have access to BIOS
- import an organisation key to BIOS
- remove the Microsoft's key for SecureBoot if Windows is not needed (preferred)

- SecureBoot in itself does not mean much
- in most Linux distributions, SecureBoot can be enabled but *initrd* is not verified
- from <https://wiki.debian.org/SecureBoot>: "*SB is also not meant to lock users out of controlling their own systems.*"
 - ... we have to use it to do exactly that
- use SecureBoot, but with:
 - a signed kernel and a signed ramdisk (UKI kernel image)
signed with a key accepted by BIOS

System security - TPM

- use TPM (Trusted Platform Module) for system integrity
- keep secret keys in TPM chip, unlocked only when booting a signed kernel
- use either:
 - disk encryption with `systemd-cryptenroll` or similar
 - in case confidentiality is not required, use some combination of `dm-verity` and/or `dm-integrity` (write operations possible only with secret keys - in case data is tampered, read operations trigger I/O errors)
- this means that evil cracker who boots mounts the system from another OS/machine drive can try to modify the system, but it will be recognized as corrupt by the installed system

System security - root access

- prevent users from getting root
 - before this, we have on *purpose* allowed root for users on machines used only by a single person, because of a matter of principle (users should have control over the machines they use)
- fix all security bugs leading to privilege escalation to root 😊

The examination mode - how to implement?

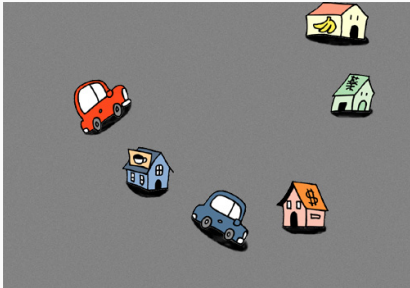
- in addition to system security, we also need the examination mode
- how to lock the student out of their normal programs and files during the exam session?
 - use a distinct user account for exam session
 - use a distinct Linux virtual terminal with its own Xorg/Wayland session (and prevent VT switching)
 - lock user interface so that only a very limited functionality is available (for example GNOME needs customization and patching to achieve this)
 - use `bwrap` to setup temporary directories shadowing access to shared directories such as `/tmp`
 - make sure there no globally writable files or directories accessible from the exam session

The examination mode - TOMOYO for MAC?

- use some Linux mandatory access mechanisms for stricter application locks
- TOMOYO Linux is one promising choice: it allows locking a desktop session process tree so that all processes in that (exam) session
 - need to have explicit permissions to run other programs (specifying which programs are permitted)
 - need to have explicit permissions to access network (specifying which network addresses)
 - may have other restrictions such as controlled filesystem access
- in TOMOYO, processes outside the restricted process tree (normal mode) need not be affected

from TOMOYO Linux documentation:

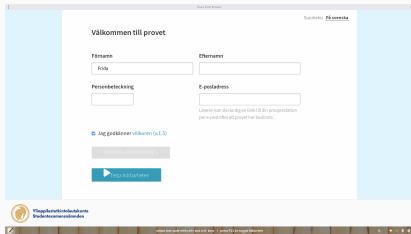
Unrestricted access:



Restricted access using MAC:



The examination mode - custom web browser



- exams will likely use web technologies to implement user interfaces and server access
- write your own web browser with some scripting language, use webkit as the browser engine, and in the UI provide only the features needed to take exams
 - for example: no accessible url bar

The examination mode - surveillance?

we could also:

- provide a means of realtime exam client monitoring (e.g. Veyon)
- spy on user during the exam session:
 - log user keystrokes
 - take screenshots periodically
 - send these to a remote server to catch possible cheaters

Really?

- the questions in "Can You Trust Your Computer?" by Richard Stallman are relevant here
(<https://www.gnu.org/philosophy/can-you-trust.html>)
- the issues surrounding Digital Rights Management are similar
- does this violate GPL version 3?
 - my understanding is that this is permitted as long as the computers are owned by schools
 - schools have the right to manage their systems as they like but they probably do not need or want to subvert the operating system security in this case

Really really?

- the system requirements of the upcoming Finnish upper secondary school digital exam application are not told to us explicitly
- that we need these features is in part only my conjecture
- but as we know that Windows, Mac OS and Chrome OS are explicitly supported and free software systems are not, we need some similar level of system and exam mode security
- at least in theory any weakness in the operating system in related functionality can put us into "not to be used in schools"-category

Puavo OS implementation

- Puavo OS (Debian) is running on maybe about 20% of the Finnish upper secondary school laptops
- if we do not react to this we will eventually lose that market to proprietary alternatives... unless politics or reality intervenes
- in Puavo OS these features are currently very much a work-in-progress and will take some time to implement
- the good news is that with Linux and free software, we can probably do this much better than the alternatives, as we can customize everything just like we want to ("we can build better cages")

A word of caution

A word of caution to educators in other countries:

- consider the usefulness of written examinations
- consider the usefulness of pen and paper for written examinations
- in case you really want digital examinations:
 - consider using a separate set of computers for those
 - you will really want a policy of open/unlocked devices for schools (choosing proprietary tech does affect our computing culture)

Thank you for listening!