# *Five ways to hack and cheat with Bring Your Own Device (BYOD) electronic examinations*

**Phillip Dawson**

Phillip Dawson is a Lecturer at Monash University. His research focuses on how university teachers use different sorts of evidence to inform their teaching. Address for correspondence: Dr Phillip Dawson, Office of the Vice-Provost (Learning and Teaching), Monash University, PO Box 197, Caulfield East, Victoria, 3145, Australia. Email: phillip.dawson@monash.edu

## Abstract

Bring Your Own Device electronic examinations (BYOD eExams) are a relatively new type of assessment where students sit a face-to-face exam under invigilated conditions on their own laptop. Special software restricts student access to prohibited computer functions and files, and provides access to any resources or software the examiner approves. In this study, the decades-old computer security principle that 'software security depends on hardware security' is applied to a range of BYOD eExam tools. Five potential hacks are examined, four of which are confirmed to work against at least one BYOD eExam tool. The consequences of these hacks are significant, ranging from removal of the exam paper from the venue through to receiving live assistance from an outside expert. Potential mitigation strategies are proposed, however these are unlikely to completely protect the integrity of BYOD eExams. Educational institutions are urged to balance the additional affordances of BYOD eExams for examiners against the potential affordances for cheaters.

## Practitioner Notes

What is already known about this topic

- Bring Your Own Device electronic examinations (BYOD eExams) allow students to sit an invigilated exam on their own laptop in an exam hall
- BYOD eExams provide a range of additional affordances not available in pen-and-paper exams, such as rich media and specialist software
- Examiners can choose which features of the student's computer they wish to allow, and 'lock down' everything else

What this paper adds

- The computer security principle 'software security depends on hardware security' is applied to BYOD eExams

- Four attacks against BYOD eExams are confirmed to work, and one attack is theoretically possible but untested
- These attacks provide a range of potential cheating options for students, including: bringing their study notes into the exam; having live access to an external expert; and removing the exam paper from the venue.

Implications for practice and/or policy

- Potential approaches to mitigate against the attacks are presented
- Educational institutions should balance the risks to exam integrity against the additional affordances of BYOD eExams
- In circumstances where exam integrity is paramount, BYOD eExams may not be an appropriate choice

## Introduction

Face-to-face invigilated written examinations hold a special place in education, and particularly in higher education. The exam is regarded by some as a particularly trustworthy type of task, representing "continuity and stability" (Carless, 2009, p. 82). Although critiques of examinations from a variety of perspectives abound (for example Biggs, 1999; Carless & Lam, 2012; Nelson & Dawson, 2014), we can say with relative certainty that we know who has undertaken an exam and the circumstances they were in. Recently a trend has emerged to allow students to bring their laptops into the exam hall and conduct the exam through special software that restricts access to only certain computer functions. This article uses computer science principles and proof-of-concept attacks to argue that student-provided hardware negates any perceived trustworthiness provided by exam hall invigilation.

The sort of Bring Your Own Device Electronic Examination (BYOD eExam) this article is concerned with occurs in an exam hall, under the supervision of an invigilator or proctor, on hardware that the student has brought with them. Examples of this sort of approach can be found in Europe (Digabi, 2014), Northern America (ExamSoft, 2014) and Australia (Transforming Exams, 2014). In a BYOD eExam, the examiner can configure the software to allow students access to certain computer features (such as calculator or dictionary tools) and deny access to others (such as parts of the Internet). They have a number of advantages over paper-based examinations. An examiner can provide students with rich media, such as an x-ray for a medical imaging task, or an audio file for a foreign-language test. The substantial portion of students who think they type faster than they write (Mogey, Cowan, Paterson, & Purcell, 2012) can type their exam responses. It is even possible for parts of the test to be computer-marked before the student leaves the exam hall.

Bring Your Own Device eExam vendors are somewhat aware of potential threats to exam integrity from hacking. Historical data from online exam hacking forums tells a story of one commercial vendor, ExamSoft, learning of hacks and finding ways to not only protect against them, but also to detect and prosecute hacking. They also appear to have engaged in an online public relations campaign to change public perception about cheating in eExams. The open-source community, particularly the Finnish Exam board, appears to have a different philosophy about hacking, even going to the extent of holding hacking competitions (Digabi, 2014). But even their best efforts are undermined by the basic computer security principle that "All software security depends on hardware security. If the hardware can be stolen or surreptitiously replaced, secure software will not help." (Barkley, 1994) When we invite students to bring hardware we do not control to the exam, we cannot claim with certainty that the software we ask them to run will perform as we expect.

Rather than privately disclose these vulnerabilities to BYOD eExam vendors, this article makes a full public disclosure (Schneier, 2007), so that all involved in decision-making about this class of tools are informed. The full disclosure philosophy argues that it is very unlikely that a single security researcher is the only person aware of a particular vulnerability, and that the best way to force a vendor to fix a problem is to make it public. This paper does not however reveal which tools are vulnerable to which attacks, nor does it provide a 'how-to' for potential cheating students.

Given that computer security principles suggest that, in theory, BYOD eExams may be insecure, this paper investigates the following research question:

*What types of attack are BYOD eExams vulnerable to, and what is the potential effect of those attacks?*

Before outlining the different types of attack, some background technical explanation is necessary, which is provided in the next section.

## A brief technical primer

The BYOD eExam tools explored in this paper fall into two categories: those that run as a program on a student's existing operating system; and those that boot (usually from a USB drive) into a completely separate operating system.

### *eExams that run on a host operating system*

Some BYOD eExam tools are programs that provide an examination environment within an existing installation of Windows or Mac OS. They attempt to 'lock down' the student's computer and prevent unauthorized use of hardware, software, files, network and other computer functions. One tool in this category is ExamSoft, which is used to administer examinations for university courses and entrance into professional societies. Although these sorts of approaches appear to be secure, it is difficult to know how much cheating is happening in secret. Parallels can be drawn to cheating in computer games: developers do their best to detect cheat programs (such as 'aimbots' that help players aim in shooting games) but some gamers pay substantial sums of money to secretly exploit vulnerabilities the developer is not yet aware of (Wendel, 2012). Without complete control over a computer – such as university-owned computer laboratories – it is not possible to control exactly what software is running on a machine (Barkley, 1994).

### *Live booting operating systems*

When a computer is booted it looks for an operating system, such as Windows, Linux or Mac OS. This usually exists on the computer's internal hard disk, but most computers can be configured to first look for an operating system on an external USB device instead. Versions of all major operating systems have been produced which can run entirely off a USB disk. BYOD eExam tools that use a live booting method run a customized, locked down version of the operating system, configurable by the examiner. It is possible to give students an operating system that disallows access to their computer's hard disk; limits their access to the Internet; or provides them with only certain programs to use.

Similar approaches are used in the corporate world for some employee BYOD programs. Individuals purchase their own laptop, and to use it at work they boot from a USB drive provided by their employer. However it is possible to set a computer up such that it does not do what the USB BYOD environment expects. Advocates of BYOD approaches use caution when expounding the security of BYOD approaches for the corporate world (James & Griffiths, 2012).

## Five attacks against BYOD eExams

In this section the five attacks against BYOD eExams are described. These have been broken into two groups: those attacks that have been confirmed to work with one or more eExam tools by myself or other researchers, and those attacks that are at this stage theoretical. They are roughly ordered by degree of difficulty for a student to independently implement the attack. Table 1 provides a summary of the five attacks, which are then expressed through vignette and an explanation.

*Table 1: Summary of the five attacks*

| *Attack* | *Status* | *Implications* | *Skill required to develop* | *Skills required by student in exam hall* | *Potential mitigation strategies* |
|---|---|---|---|---|---|
| Copying contents of USB to hard disk | Confirmed | Student takes exam paper out of exam hall; copy of eExam software is made available | Copying files | None: computer can be configured to run the attack invisibly when the USB drive is inserted | Do not distribute exam paper on USB |
| Virtual machine | Confirmed | Student has complete control of computer, can access prohibited files, programs or the Internet | Using virtualization software | None: computer can be configured to run the attack invisibly when the USB drive is inserted | Probe operating system for virtual-machine-specific functionality |
| USB keyboard | Confirmed | Student has access to any | Editing a text | None | Disable USB |

| | | | | | |
|---|---|---|---|---|---|
| hacks | | text, eg notes | file | | keyboards |
| Modifying software | Confirmed | Student has complete control of computer, can access prohibited files, programs or the Internet | Programming | None: computer can be configured to run the attack invisibly when the USB drive is inserted | Make substantial changes to software between each exam |
| Cold boot attack | Theoretical | Student takes exam paper out of exam hall; copy of eExam software is made available | Substantial knowledge of low-level computer hardware and programming | None | Mandatory waiting period after exam finishes before students can leave venue |

*Confirmed attacks*

### Copying contents of USB to hard disk

*A student walks into the examination hall, takes out her laptop, opens it up, and puts in the USB key when instructed to. Her computer appears to boot up and function identically to her peers' – but when she leaves the exam she takes a copy of the exam paper with her.*

Some BYOD eExam tools use the USB key for distributing the exam software and the exam paper. If the contents of the USB key leave the exam hall, then this includes the exam paper and the eExam software. An unsophisticated version of this attack would simply involve the student dragging and dropping the files from the USB to their own computer. A student who was afraid of being caught by an invigilator could instead use a script (saved sequence of commands) that automatically copies the disk when it is inserted, all while displaying a blank screen as if the computer was off, then automatically reboots and runs the exam software.

An obvious solution to this attack is to not distribute the exam paper via USB, and that is the approach taken by Digabi and Examsoft. Examiners who wish to keep their exam papers 'secret' (perhaps so they can reuse parts each year) would be wise to avoid tools that include the exam paper on a USB given to students.

### Virtual machine

*Another student walks into the exam hall with a laptop that looks like any other, but when the exam software 'boots' it is actually running inside a virtual machine on another operating system. He puts his headphones on, ostensibly to listen to a* *multimedia portion of the paper. In the background he is running a one-way Skype call and sharing his screen over a cellular Internet connection. His exam coach on the other end takes a look at the exam paper and starts dictating…*

Virtualisation software allows a user to run one operating system inside another through the creation of a Virtual Machine (VM). An example use of this is to run Windows inside the virtualization tool Parallels on a Mac running OSX, which is a common approach to providing access to Windows-only software to Mac users. A student could use this same technology to run the eExam software within a VM, all while retaining the affordances of a full operating system running in the background.

An unsophisticated version of this attack would involve the graphical interface of a free virtualization tool; this takes roughly the same amount of time as booting directly into the eExam tool. A student concerned about being seen by an invigilator using a virtualization tool could script a more sophisticated version of this attack: show a blank screen, wait for the USB to be inserted, then automatically run the VM software, which appears the same as a normal boot sequence. Creating this script is within the grasp of a typical information technology student.

By having access to the operating system running in the background, a student has access to all of the features the eExam system is trying to hide: files, the Internet and any prohibited programs such as screen sharing or chat tools.

One eExam vendor has made particular efforts to combat against this attack by having their software refuse to run if it detects it is operating within a VM. However, it is theoretically – and increasingly, practically – possible to run an operating system inside a VM that is transparent. Ironically, creating an undetectable VM environment that appears exactly the same to the software running inside it as 'bare metal' hardware is a pursuit of the computer security community. Researchers who investigate computer viruses use VMs to quarantine malware for study. This has prompted malware authors to deactivate their malware when running on virtual machines, to thwart attempts at analysis. As computer security researchers pursue the undetectable VM for studying viruses, they may also assist dishonest students seeking to run their eExams in a VM.

This issue has been rated "Critical – The vulnerability has severe consequences and may undermine the whole project" by one computer security researcher who participated in the Digabi hacking competition (Sintonen, 2013). He further noted "There is no obvious way to prevent an attack of this kind, unless if there is some way to guarantee the integrity of the hardware. It is extremely unlikely such integrity could be guaranteed in the environment envisioned"

### USB keyboard hacks

*This student's laptop looks like any other, but one of its internal USB ports has a special device attached. The computer thinks it is a keyboard, but it is actually a 'USB key injector', which she bought for $40 online. She has stored her study notes on it, and five minutes into the exam her 'internal keyboard' will type them out instantly for her reference.*

All major eExam tools considered allow the use of external keyboards, and for good reasons: it would be torturous to type at a laptop keyboard for two or three hours at exam pace. However, computers tend to trust that any device claiming to be a USB keyboards is a keyboard. It's possible to build or buy a variety of devices that tell the OS they are USB keyboards but they are something else: remote controls, gamepads, or USB key injectors.

USB key injectors are programmed to transmit certain keypresses under certain conditions (HakShop, 2014). A simple usage in a BYOD eExam would be to type out a set of study notes into whatever text box was selected a certain number of minutes into the exam. Having a USB dongle sticking out from the computer could look suspicious, but many laptop computers also have internal USB capabilities. On some laptops, hiding this device is as simple as unscrewing half a dozen screws and plugging the device in to an exposed USB port.

Denying students access to USB keyboards is not really an option, for technical and accessibility issues. Many laptop internal keyboards now use USB to connect to the motherboard, and are thus indistinguishable from a key injector. Further, forcing students to type for hours on a small laptop keyboard risks injury or complaint. It is difficult to formulate a solution to this attack that does not involve a visual examination of the interior of every student's laptop internals.

### Modifying software

*This student boots up her laptop to the eExam software, just like her peers – except that she is running a modified copy pre-installed on her computer. Her version of the eExam software uses the same instructions to load the exam paper from the USB or network, but also includes a set of pre-made essays and some tools to hide her copy-pasting.*

Some BYOD eExam tools are open source and based on the Linux operating system. This means that it is perfectly legal to make and distribute derivative operating systems – including one that is designed to help students cheat. This type of attack would provide the same affordances to a cheating student that the VM based attacks would, however without the complication of having to hide the VM functionality from the operating system.

Creating this derivative eExam tool would take more work than just running the tools in a VM. After scouring online eExam hacking forums I am not aware of a working hack in this category that has been distributed, although Sintonen (2013) seems to have successfully executed this sort of attack against Digabi, rating it "High – The vulnerability leads to a full system compromise or other similar dire consequences".

*Theoretical (unconfirmed) attack*

### Cold boot attack

*Part way through the exam, this student's computer appears to have a hardware failure and abruptly power off, then back on again. In the time between the powering off and booting back to the exam, the student's computer has silently dumped the contents of its RAM to a file on her hard disk. At home after the exam is over, the student uploads this file to sell on the Internet to experts who extract the exam paper and sell it for a fee.*

Cold boot attacks involve extracting the contents of a computer’s RAM after it has switched off, and examining it forensically. Recent experiments conducted by Gruhn and Muller (2013) and Halderman, et al. (2009) demonstrate that cold boot attacks are effective against modern computer hardware; how-to guides and tools are also available online (Halderman et al., 2009; Rankin, 2009).

A cold boot attack on an eExam would require students to quickly turn their computer off, dump the contents of RAM to the computer’s hard disk, then boot back into the eExam tool. A competent later-year IT student could set up a laptop to perform this sequence of actions if, for instance, the power plug was pulled or the laptop battery failed. When examined by someone with the appropriate skills, an eExam RAM dump may contain the entire exam paper, or the software used to run the exam.

There are no guarantees that any one RAM dump will contain any or all of the examination, but a set of students working together may be able to assemble the entire paper. In a sense this approach is similar to the ‘brain dumps’ method used to help students cheat on certification examinations, where immediately after the exam students upload as many of the questions as they can remember to a website that collates them. Brain dump websites currently provide reasonably accurate copies of tests required for a variety of professional accreditations; perhaps RAM-dump sites may be their BYOD eExam equivalents.

## Discussion and conclusions

This investigation has revealed multiple ways that students can cheat in BYOD eExams, which collectively bypass all of the restrictions of this class of tool. The BYOD eExam is by definition less secure than both pen-and-paper examinations, and examinations held in a computer laboratory, as it has all the vulnerabilities of both environments, as well as some of its own.

Each of the main BYOD eExam vendors is aware of one or more of these potential attacks. When I have raised these attacks with proponents of BYOD eExams they have made counter-arguments that fall in the category of security through obscurity, for example: “nobody would bother doing that for my examination, as the stakes are low”. Another argument, made partially in jest, is that “if a student is clever enough to do all of that then they deserve to pass”. However, some of these attacks are not complicated to implement, and others can be bundled up into easy-to-install packages.

If BYOD eExams see widespread adoption, a black market of exam cheating software may emerge. Proactively engaging with the hacking community may prove more productive than dismissing them as a threat, or pursuing them with legal action (Schneier, 2007). In the parallel field of computer game cheating, where the stakes are much lower, there is already a lucrative market for new hacks to help gamers win. If BYOD eExams are to persist, they may need to learn from the experiences of computer entertainment companies. Computer games researchers are currently exploring technologies like Trusted Computing as a method of preventing attacks similar to the five presented in this paper (Balfe & Mohammed, 2007).

Although workarounds may be possible against particular implementations of the attacks presented in this article, they are but symptoms of a larger problem. The need to control hardware for software to be secure was identified by NIST researchers in the 1990s (Barkley, 1994), and problems with BYOD systems have been identified by independent hackers (Sintonen, 2013) and computer security researchers (James & Griffiths, 2012). Allowing students to control the hardware and initial software state of the machines they bring in to the exam is a fundamental but necessary design weakness of BYOD eExams.

So, if BYOD eExams are not actually secure, are they still a useful alternative to the pen-and-paper examination? It is worth noting that assessment always serves multiple purposes (Boud, 2000), and although this paper finds numerous problems with BYOD eExams' ability to generate rigorous grades, the eExam may be an improvement with respect to assessment's other purposes. Assessment should guide students through tasks that lead to learning; perhaps the additional affordances of the BYOD eExam may better support this than pen-and-paper. Assessment should lead to learning in the long term, which might be facilitated by careful use of online peer/self/co assessment options made available by eExams. Decisions to use one form of assessment over another are complex and currently poorly understood (Dawson et al., 2013); and despite these attacks, BYOD eExams may still be the best option in some circumstances.

## References

Balfe, S., & Mohammed, A. (2007). Final Fantasy – Securing On-Line Gaming with Trusted Computing. In B. Xiao, L. Yang, J. Ma, C. Muller-Schloer & Y. Hua (Eds.), *Autonomic and Trusted Computing* (Vol. 4610, pp. 123-134): Springer Berlin Heidelberg.

Barkley, J. (1994). *Security in open systems*: US Government Printing Office.

Biggs, J. (1999). What the Student Does: teaching for enhanced learning. *Higher Education Research & Development, 18*(1), 57-75. doi:10.1080/0729436990180105

Boud, D. (2000). Sustainable Assessment: Rethinking assessment for the learning society. *Studies in Continuing Education, 22*(2), 151-167. doi:10.1080/713695728

Carless, D. (2009). Trust, distrust and their impact on assessment reform. *Assessment & Evaluation in Higher Education, 34*(1), 79-89. doi:10.1080/02602930801895786

Carless, D., & Lam, R. (2012). The examined life: perspectives of lower primary school students in Hong Kong. *Education 3-13*, 1-17. doi:10.1080/03004279.2012.689988

Dawson, P., Bearman, M., Boud, D. J., Hall, M., Molloy, E. K., Bennett, S., & Gordon, J. (2013). Assessment Might Dictate the Curriculum, But What Dictates Assessment? *Teaching & Learning Inquiry: The ISSOTL Journal, 1*(1), 107-111. doi:10.2979/teachlearninqu.1.1.107

Digabi. (2014). Digabi OS Retrieved 24 April 2014, from https://digabi.fi/tekniikka/digabi-os/

ExamSoft. (2014). ExamSoft Provides Direct-Evidence of Student Learning Outcomes, Offline Computer-based Exams, and Real-time Feedback Retrieved 24 April 2014, from http://learn.examsoft.com/

Gruhn, M., & Muller, T. (2013, 2-6 Sept. 2013). *On the Practicability of Cold Boot Attacks.* Paper presented at the Availability, Reliability and Security (ARES), 2013 Eighth International Conference on.

HakShop. (2014). HakShop — USB Rubber Ducky Deluxe, from https://hakshop.myshopify.com/products/usb-rubber-ducky-deluxe

Halderman, J. A., Schoen, S. D., Heninger, N., Clarkson, W., Paul, W., Calandrino, J. A., . . . Felten, E. W. (2009). Lest we remember: cold-boot attacks on encryption keys. *Commun. ACM, 52*(5), 91-98. doi:10.1145/1506409.1506429

James, P., & Griffiths, D. (2012). The Mobile Execution Environment: A Secure And Non-Intrusive Approach To Implement A Bring Your Own Device Policy For Laptops. In T. Williams, M. Johnstone & C. Valli (Eds.), *10th Australian Information Security Management Conference* (pp. 100-109). Perth, Western Australia: Security Research Institute, Edith Cowan University.

Mogey, N., Cowan, J., Paterson, J., & Purcell, M. (2012). Students' choices between typing and handwriting in examinations. *Active Learning in Higher Education, 13*(2), 117-128. doi:10.1177/1469787412441297

Nelson, R., & Dawson, P. (2014). A contribution to the history of assessment: how a conversation simulator redeems Socratic method. *Assessment & Evaluation in Higher Education, 39*(2), 195-204. doi:10.1080/02602938.2013.798394

Rankin, K. (2009). Cold Boot Attack Tools for Linux. *Linux Journal*.

Schneier, B. (2007). Full Disclosure of Security Vulnerabilities a 'Damned Good Idea' Retrieved 28 March 2014, 2014, from https://http://www.schneier.com/essay-146.html

Sintonen, H. (2013). hackabi contest entry, version 1.8, from https://sintonen.fi/advisories/hackabi.txt

Transforming Exams. (2014). Transforming Exams - A scalable examination platform for BYOD invigilated assessment Retrieved April 24 2014

Wendel, E. (2012). *Cheating in Online Games: A Case Study of Bots and Bot-Detection in Browser-Based Multiplayer Games.* Norwegian University of Science and Technology, Trondheim, Norway. Retrieved from http://www.diva-portal.org/smash/record.jsf?pid=diva2:570786