ASEE's 123rd Annual • Conference & Exposition • New Orleans, LA • June 26-29, 2016

New Orleans Jazzed about Engineering Education

Paper ID #16036

Ethics in the Classroom: The Volkswagen Diesel Scandal

Dr. Elisa L. Warford, University of Southern California

Elisa Warford is a senior lecturer in the Engineering Writing Program at the University of Southern California, where she teaches undergraduate and graduate courses in written and oral engineering communication. Her current research interests include the rhetoric of science and portrayals of engineering and technology in American literature. She is also a professional technical editor specializing in engineering writing for academia and industry. She holds a Ph.D. in English from the University of Maryland.

Ethics in the Classroom: The Volkswagen Diesel Scandal

Abstract

Given ABET's criteria for ethics education and the professional responsibility of engineers to uphold reasonable standards of ethics, some form of ethics education in the engineering curriculum has become standard in engineering schools. While a microethics case study approach is common in the classroom, some researchers have called for greater emphasis on a macroethics approach, which emphasizes broader ethical questions about technology in society rather than questions about individual professional responsibility. This paper presents a case study, written by the author, used to discuss a macroethics issue related to the 2015 Volkswagen diesel scandal. The case study uses the Volkswagen case to highlight the possible ethical dilemma between the need for greater public and researcher access to vehicle software code versus the automobile manufacturers' desire to keep the code proprietary. The paper applies the utilitarian, justice, virtue, and rights frameworks to the case, and shows how students can use the frameworks to analyze a range of ethical dilemmas and to better understand the broad ethical implications of open-source software, including questions about software vulnerability disclosures, intellectual property, and public safety. The paper presents an excerpt from a student paper demonstrating the learning outcomes of the case study exercise. The full text of the case study is included as an appendix.

Introduction

Since ABET's 2000 requirement for an ethics component in engineering education,[1] instruction in ethics is now commonplace in engineering curricula. The 2015 Volkswagen diesel scandal, however, has sparked calls for moving away from "this compliance mindset" in ABET accredited schools to one in which ethics instruction is integrated more effectively and thoroughly into the engineering curricula.[2] There is considerable debate about the most effective way to incorporate ethics instruction in the classroom, whether as a stand-alone course or as incorporated throughout the engineering curriculum (or both). There have also been calls for greater emphasis on a macroethics approach over a microethics approach.[3-5] Macroethics, as Herkert defines it,[3] considers broad ethical ethical questions about the role of engineering and technology in society, while a microethics approach focuses on questions about the ethical behavior of individual engineers or institutions and on concerns regarding professional responsibility.

Each of these approaches, micro and macro, typically uses different classroom techniques to address its questions. In microethics instruction, case studies are widely used, such as well known historical engineering failures such as the Bhopal chemical leak, the Challenger shuttle disaster, or the Kansas City Hyatt walkway failure.[6] In these case studies, the decisions of engineers and managers are scrutinized to determine their ethical missteps. Numerous case studies involving professional responsibility can also be found on the NSPE website, which publishes cases brought before its ethics board along with the board's analysis.[7] Case studies have been shown to be useful ways of teaching questions about professional responsibility, as they often present complex real-world dilemmas that are morally ambiguous and may have multiple valid responses,[8] encouraging students to draw on their engineering problem-solving skills in a new way.

One drawback of case studies, however, is that they usually emphasize microethics issues over macroethics issues,[8] that is, issues of individual professional responsibility over the socio-technical systems in which technology is designed. Johnson and Wetmore argue that by focusing on the individual responsibility of the engineer, it is easy to lose "the idea that interest groups corporations, governments, professional associations, and others are essential to good engineering outcomes."[4] They therefore call for a greater incorporation of Science and Technology Studies scholarship in engineering ethics. Macroethics pedagogical approaches are also typically more closely associated with STS. Riley,[5] for example, calls for a macroethics pedagogy that adopts pedagogical methods from the social sciences and philosophy. She argues that macroethics approaches often "do not fit as neatly into modes of engineering thought" of straightforward, right-or-wrong problem-solving and that engineering ethics instruction should challenge students to question social structures of power and authority.[5] Her courses thus incorporate readings from a wide range of disciplines and call on students to lead class discussions and write reflective papers about their learning process. Herkert also calls for new course materials to be developed and published, such as case studies related to macroethics issues.[3]

In answer to these calls for increased instruction in macroethics issues, this paper describes my endeavor to incorporate a case study approach into a macroethics unit in an advanced writing course for engineers. In this unit, I use a case study to introduce and discuss the macroethics implications of the 2015 Volkswagen diesel scandal, in which Volkswagen installed software that allowed several models of its diesel automobiles to pass government emissions tests while emitting higher levels of nitrous oxide on the road. The paper first provides background on the ethics unit in the course, then introduces the case study and the way it is used in class. The paper then discusses some of the ethical issues raised by the Volkswagen case using the utility, rights, justice and virtue ethical frameworks. As assessment data, an example is provided of the way one student applied the class discussion generated by the case study to his ethics paper. The full text of the case study is included as an appendix to the paper.

Engineering ethics in an advanced engineering writing course

At the University of Southern California Viterbi School of Engineering, the ethics component of the engineering curriculum is satisfied in large part by an ethics unit the Writing 340 course, Advanced Communication for Engineers. Writing 340 is an upper-division course required of all undergraduates, who choose from several versions of the course according to their major field of study. Over 90% of engineering students take the engineering version. In part because the instructors of these courses are not engineers with practical experience in industry, most of the instructors adopt a macroethics approach in their classes. The students are required to write a position paper about an ethical issue faced by the profession, such as sustainable development or energy, online privacy and security, or human enhancement. They are encouraged to write about a question in their particular field in engineering but are not required to. In my classes, students write a paper similar to articles that might be found in academic journals such as *Science and Engineering Ethics* or *IEEE Technology and Society* directed to an academic, professional audience.

To prepare students for the paper, I introduce basic moral theories in lecture and through handouts.[9-12] I introduce utilitarianism, rights theory, justice theory, and virtue ethics. I choose these theories because they are widely used and applicable to a wide range of ethical topics, although other theories, such as care ethics, might also be used. I also introduce the professional codes of ethics and discuss their underpinnings in moral theory. For example, the primary canon of all codes, which states that the engineer should work for the benefit of the public,[13] can be viewed through the lens of two moral theories: utilitarianism, in that the health and safety of the many outweigh the financial gain of the few; and rights, in that the public has a human right to health and safety.

To illustrate to the students how these moral theories might be applied to macroethics issues in engineering, I adopt a case study approach. The Volkswagen diesel scandal of 2015 provides a provocative, timely controversy involving engineering macroethics.[*] Briefly, the case asks whether, given the increasingly complex software in vehicles, which controls virtually every system in the car, auto manufacturers' proprietary software code should be made available to car owners and security researchers so that they can review and/or modify it. The code is currently protected under the Digital Millennium Copyright Act (DCMA), which allows auto manufacturers to limit its availability to authorized dealers and repair services only. The case study in its entirety can be found in the appendix to this paper.

The case study can be used in the classroom in a single period of 75 minutes (not including the introductory lecture on moral theory). The case study follows the principles of effective case studies found in Dyrud and Sharp[8] in that it

1. Accommodates the students' level of knowledge on the topic. The students in this course come from a variety of disciplines, within and outside of the school of engineering. The case provides sufficient background information to allow all students to participate in the discussion. Students without backgrounds in software engineering are given enough information to discuss the topic, while students with more knowledge about coding can enhance the discussion with their expertise.

---

[*] It may well be that in time it will also provide an interesting case for study as a microethics issue—who decided to add the defeat device to the engines, who implemented it, and who knew about it and gave approval? At the time of this writing, several engineers have been suspended, but it is unclear who was actually responsible, and what kind of institutional failure allowed such a plan to be implemented.

2. Is simple to understand yet is complex enough to engage students in the problem solving process.
3. Allows for more than one right answer and viewpoint.
4. Can be discussed in one 75-minute class period.
5. Is not overly long or complex.
6. Allows "the instructor to prepare without feeling overloaded."[8]

I divide the students into groups, give them a few minutes to read the case, and have them follow the instructions written on the case study. They discuss the case in their small groups, and then we reconvene to discuss the case as a class.

Ethical analysis

The central issue in the case study is whether vehicle code software should be made open to the public and/or software researchers, or remain proprietary under the Digital Millennium Copyright Act (DMCA). This issue also raises questions about the ethics of open source software in general. I encourage the students to approach the case from a number of different moral theories, and I encourage them to refer to the moral theory handouts I introduced earlier. As the analysis below shows, the case raises complicated ethical issues without a clear-cut answer. This analysis is not meant to be a comprehensive ethical analysis of this issue, but is offered rather to introduce some of the ways discussion might be directed in the class.

Utility

As directed by the case study, the students are encouraged to consider the effects of different alternatives—open source code[†] vs. proprietary—on various stakeholders and then to determine which alternative results in the greatest good for the greatest number of people. The main utilitarian argument of the open code advocates would be that by making the code accessible to the owners and security researchers, deceptive acts like Volkswagen's could be caught sooner. While relatively few people would have the time, skills, or motivation to comb through millions of lines of code, some presumably would. In the case of the Volkswagen scandal, this would mean better health for the public and the environment, in that the affected cars could be modified or taken off the road sooner, reducing harmful emissions. In addition, security vulnerabilities could be discovered and patched sooner, preventing hackers from exploiting them. Ostensibly, having more "eyes on" the code would also reduce the number of bugs in the software, which could reduce other safety risks due to flawed software. All of these results would benefit the

[†] I recognize the distinction between open source software and free software, as distinguished by Richard Stallman,[14] who argues that open source software advocates focus too strongly on the pragmatic advantages of open source rather than on the moral right of people to see, modify, and redistribute the software they use. However, for the purposes of this case study, which I use in a general education course with students from many different majors, I generally conflate the two rather than making the distinction in discussion. Perhaps it would be appropriate to do so in a computer science course.

general public, which would be safer and enjoy better health. Open code may also benefit car enthusiasts, who would have greater freedom to personalize or repair their cars and enjoy their pastime. The Electronic Frontier Foundation argues that auto hobbyists have also made modifications to increase fuel efficiency and implement new vehicle functions, modifications that may be adopted in future car models,[15] which would also benefit the general public. It is also possible that making vehicle code open would create a new industry of third-party software companies which could modify or add to existing code, allowing owners without coding skills to modify their cars as they desired.

Open source software advocates also argue that open source increases the quality of the software in general, due to greater collaboration, which would also benefit the general public. However, the claim that open source software is generally high quality has been disputed.[16] Moreover, it can be argued that allowing car owners the freedom to modify their software would also enable them to skirt air quality regulations, just as Volkswagen did, thereby putting the health of the public at risk. Indeed, this is the stance that the EPA has taken. The EPA argues that allowing owners and security researchers access to the code would allow owners to "increase power and/or boost fuel economy," which "will often increase emissions from a vehicle engine."[17] However, the number of enthusiasts capable and interested in doing this would probably be small compared to the number of people who could benefit from greater overall security. The DOT has also raised concerns that if security researchers are allowed access to vehicle code, they may publish information about security vulnerabilities before properly informing the manufacturer. If the manufacturer were not given enough time to fix the flaw, this could put the public at risk of hacking attacks.[18] However, so far security researchers seem to be sensitive to this issue: in the Jeep hack event referenced in the case study, the researchers disclosed the information to GM nine months in advance of publishing the information. When it was published, they omitted details about how they rewrote the firmware.[19] Other researchers have cooperated similarly with auto manufacturers.[18]

Open source software advocates argue that open source software would benefit humanity overall by providing greater access to information. Alternatively, open source software might discourage innovation, as car manufacturers would argue, because it would reduce the financial incentive to develop innovative technology. Car manufacturers argue that they would suffer from making their code public, as their investment into creating the code would no longer be proprietary and could be copied by their competitors.

Rights

The car manufacturers may argue that making their code public would infringe on their right to own (and profit from) their intellectual property. Indeed, this is the motivation of the code's protection under the DMCA, and there are significant commercial interests at stake. However, open source advocates would argue that the public's right to safe products, which may be achieved by having more "eyes on" the software, outweighs the right to intellectual property. In addition, according to free software advocates such as Richard Stallman, people have a right to "free" software: they should have the freedom to run it, modify it, redistribute it, and distribute modified copies of it.[20] "Free" software is not necessarily no cost; thus Volkswagen could still charge for its software but grant its users the right to modify it. Another right at stake here is the right to privacy: as vehicles become increasingly networked, some owners may wish to protect their right to privacy by disabling data collection functions on their automobiles.[15]

Justice

Justice ethics do not play a large role in the debate over open source software. But clearly, Volkswagen's act was not fair, as it put other diesel car manufacturers, who used a more expensive diesel emissions system, at a competitive disadvantage. Also, the public bore the burden of poorer air quality, while Volkswagen profited from deceiving the public. On the other hand, one could ask whether it is fair to expect companies to develop software from which they stand to make little financial gain.

Virtue

Virtue ethics do not play a large role in the direct question of whether Volkswagen's software code should be open or closed, although Volkswagen, in antithesis to the image it tried to portray in its television commercial depicting engineers sprouting angels' wings, clearly did not act virtuously in deceiving regulators and the public with its the defeat devices. In fact, the company has suffered plummeting stock prices and disappointing sales due to consumers' loss of trust in the corporation.[21]

It could be argued that open source software is generally more virtuous than proprietary software, in that it is transparent, collaborative, and more decentralized than proprietary software, and thus better reflects democratic virtues. Free software advocates, indeed, argue that it is more in line with a free society: "As our society grows more dependent on computers, the software we run is of critical importance to securing the future of a free society. Free software is about having control over the technology we use in our homes, schools and businesses, where computers work for our individual and communal benefit, not for proprietary software companies or governments who might seek to restrict and monitor us."[22]

Solutions

As can be seen from the brief ethical analysis above, different ethical tests on this issue yield different answers to the question, and even within each ethical test, compelling arguments can be made for either side. At this point, students must evaluate which ethical framework should carry the most weight in their decisions. The analysis also reveals that many stakeholders are involved in this issue—individual engineers, corporations, the public, government agencies and non-governmental organizations—demonstrating the complex interrelationships between the practice of engineering and society. I encourage students to try to arrive at some sort of creative solution to the problem that would at least partially satisfy all stakeholders. Some students argue that the code should be made completely open source, citing the supposed practical benefits and ethical superiority of open source software. Others argue the other extreme, that the auto manufacturers have the right to keep the code proprietary. Still others argue for a compromise solution. One such solution might be to keep the software proprietary but to create a software research team at the NHTSA or EPA that had access to the code. However, given the enormous size of the code, the resources of a single government organization are not likely to be sufficient for such a project. Other solutions involve mandating a procedure that independent or academic security researchers must follow in disclosing their findings.[18] In our discussions of solutions, I emphasize the economic, social and political constraints that would need to be considered.

Assessment

Given the recentness of the Volkswagen scandal at the time of this writing, I have had the opportunity to use this case study in my classes only two times, and thus I do not have a large sample size from which to evaluate the effectiveness of the activity. Furthermore, data-driven assessment results such as multiple choice exams are perhaps not useful ways to evaluate ethics instruction.[5] Although this case study could be used for extended discussion and analysis of greater depth, the assignment serves to introduce students to the different types of arguments that can be made. The activity did elicit lively, thoughtful class discussion. The end result of this activity, along with other activities in the ethics unit of the course, is a paper on an engineering macroethics issue of the students' choosing. Most students did not write directly about this topic, but the exercise inspired one student to write about open source software in general, and he used this case in his paper as an example of an ethical failure. I quote a portion of the paper here:

> Another failure of closed source is that it allows companies to be disingenuous with their customers, as was seen with the recent Volkswagen emission scandal. Engineers at VW wrote code into the cars' software that made the car to operate within pollution guidelines when the car detected EPA testing conditions, but, when outside of these conditions, allowed the car to pollute up to 38 times the EPA limit for nitrogen oxide pollution.[23] A study has found that this could lead to the deaths of 59 Americans per year through increased air pollution.[24] If automotive code was open source, deceptions like this would be nearly impossible. The current leading argument against open source code for cars is that users could potentially install software of their own, potentially leading to dangerous effects. However, individuals have been flashing (the term for rewriting a car's ROM chips) car software already, with entire companies like SCTFlash's business model, based entirely on flashing new software into car computers.[25] Also, the U.S. Supreme Court has already stated that [it] is "fair use" for a user to install whatever software he/she wishes on purchased hardware, and that there is no violation of copyright law when done.[26] The VW deceit exemplifies that ethically, better profit margins should never trump fair use, the virtue of honesty, and the human right to safety.

The paragraph demonstrates that the student understands "the intricate ways in which the technical and social are intertwined"[27] and is able to construct his own ethical position based in part on the moral theories we introduced in class. The student appeals to several of these theories —justice, virtue, and rights—but also to legal precedent and government agencies such as the EPA. If one of my objectives for my students is to "recognize the values that influence their work, the ways their work influences values, and the other actors involved in this process,"[4] this case study exercise seems to be an effective means to this end.

Conclusion

This paper has discussed the ways ethics instruction is incorporated in an advanced writing course for engineers at the University of Southern California. It also presented a case study for use by instructors who teach engineering ethics. The case study represents a microethics pedagogical approach to a macroethics issue, the Volkswagen diesel emissions scandal. The paper also provided an introductory analysis of the ethical issues of this case using the utility, rights, justice, and virtue approach. It showed an example of the way one student successfully incorporated the discussion generated by the case study into a paper on open source software. The discussion that the case study generates hopefully helps teach future engineers about the ways that their work has serious ethical implications and consequences in society.

References

1. ABET. (2016) *Criteria for engineering programs, 2016-2017*. Retrieved from http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2016-2017/#outcomes
2. Patel, P. (2015, 25 Sep.) Engineers, ethics, and the VW scandal. *IEEE Spectrum*. Retrieved from http://spectrum.ieee.org/cars-that-think/at-work/education/vw-scandal-shocking-but-not-surprising-ethicists-say
3. Herkert, K. (2005). Ways of thinking about and teaching ethical problem solving: Microethics and macroethics in engineering. *Science and Engineering Ethics*, 11, 373-385.
4. Johnson, D. and Wetmore, J. (2008). STS and ethics: Implications for engineering ethics. In E. Hackett, O. Amsterdamska, M. Lynch and J. Watchman (Eds.) *New Handbook of Science and Technology Studies*, (567-581) Boston: MIT Press.
5. Riley, D. (2008). Ethics in context, ethics in action: Getting beyond the individual professional in engineering ethics education. *ASEE Annual Conference Proceedings*.
6. Texas A&M Department of Civil Engineering. (n.d.) *Ethics*. Retrieved from http://ethics.tamu.edu
7. NSPE. (n.d.) *Board of Ethical Review Cases*. Retrieved from http://www.nspe.org/resources/ethics/ethics-resources/board-of-ethical-review-cases
8. Dyrud, M. and Sharp, J. (2015). Ethics for beginners. *ASEE Annual Conference Proceedings*.
9. Hamilton, J. (2012). How to use the best outcomes or utility test. *Markkula Center for Applied Ethics*. Retrieved from http://ethicsops.com/UtilityTest.php.
10. Hamilton, J. (2012). How to use the justice test. *Markkula Center for Applied Ethics*. Retrieved from http://ethicsops.com/JusticeTest.php.
11. Hamilton, J. (2012). How to use the character or virtue test. *Markkula Center for Applied Ethics*. Retrieved from http://ethicsops.com/CharacterVirtue.php.
12. Hamilton, J. (2012). How to use the rights test. *Markkula Center for Applied Ethics*. Retrieved from http://ethicsops.com/RightsTest.php.
13. NSPE. (n.d.). *Code of Ethics*. Retrieved from http://www.nspe.org/resources/ethics/code-ethics
14. Stallman, R. (2016). Why open-source misses the point of free software. *GNU Operating System*. Retrieved from http://www.gnu.org/philosophy/open-source-misses-the-point.en.html
15. Walsh, K. (2014, 4 Nov.). EFF to Librarian of Congress: Let car owners look under the hood. *Electronic Frontier Foundation*. Retrieved from https://www.eff.org/deeplinks/2014/11/eff-librarian-congress-let-car-owners-look-under-hood
16. Mayer, B. (2001). The ethics of free software. *Dr.Dobb's*. Retrieved from http://www.drdobbs.com/the-ethics-of-free-software/184414581

17. EPA. (2015, 17 Jul.). *Letter to General Counsel and Associate Register of Copyrights*. http://copyright.gov/1201/2015/USCO-letters/EPA_Letter_to_USCO_re_1201.pdf
18. Kravets, D. (2015, 30 Sep.). Should research on vehicle software be hidden from the public? *ArsTechnica*. Retrieved from http://arstechnica.com/tech-policy/2015/09/should-private-research-on-vehicle-software-be-hidden-from-the-public/
19. Greenberg, A. (2015, 21 Jul.). Hackers remotely kill a Jeep on the highway—with me in it. *Wired*. Retrieved from http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/
20. Stallman, R. (1999) The GNU operating system and the free software movement. In DiBona, C., Ockman, S. and Stone, M. (Eds.) *Open Sources: Voices from the Open Source Revolution*. Sebastopol, CA: O'Reilly.
21. CBC/Radio Canada. (2016, 19 Jan.). Volkswagen shareholders to sue over plunging stock price. Retrieved from http://www.cbc.ca/news/business/volkswagen-lawsuit-1.3408862
22. Free Software Foundation, (n.d.). *About*. Retrieved from fsf.org/about
23. Russell, K., Guilbert, G., Keller, J., and Watkins, D. (2015, 22 Oct.). How Volkswagen got away with diesel deception. *New York Times*. Retrieved from nytimes.com/interactive/2015/business/international/vw-diesel-emissions-scandal-explained.html?_r=0.
24. Barrett, S. et al. (2015). Impact of the Volkswagen emissions control defeat device on US public health. *Environmental Research Letters*, 10. Retrieved from http://iopscience.iop.org/article/10.1088/1748-9326/10/11/114005/pdf.
25. SCTFlash. (2015). SCT. Retrieved from www.sctflash.com/index.php.
26. Kravets, D. (2015, 26 Jul.) U.S. declares iPhone jailbreaking legal, over Apple's objections. *Wired*. Retrieved from http://www.wired.com/2010/07/feds-ok-iphone-jailbreaking/.
27. Johnson, D. and Wetmore, J. (2004) Engineering ethics: Integrating STS and practical ethics. *Presentation to Annual Meeting of 4S, Paris*.

Appendix

Case Study in Software Engineering Ethics: Software Vulnerability Research

The Case

In September 2015, Volkswagen admitted to installing "defeat devices" on its diesel engines that misrepresented the levels of harmful emissions the automobiles were emitting on the road. The device would shift the engine into an operating mode that emitted acceptable levels of nitrous oxide, a gas that is converted to unhealthful ozone, when the car sensed that it was on a testing platform. When the car drove on the road, the engine would shift its operating mode so that the car had better fuel efficiency and increased power at the expense of higher emissions, which were up to 40 times higher than allowed by the EPA. The defeat device was installed on an estimated 11 million cars in the U.S. and Europe.

The defeat devices were based on software code, which is becoming increasingly complex in automobiles. Late model, high-end cars can have 100 million or more lines of code [1], a high number even compared with the number of lines of code for Facebook (50 million) or the Large Hadron Collider (60 million) [1]. Most of this code is contained in the car's electronic control unit, which controls steering, throttle, braking, and electrical systems in the car.
Errors in software code can be a threat to safety: Ford has recalled 432,000 cars because of faulty code that could cause the engine to keep running even after the driver had turned it off. Toyota recalled 625,000 hybrid cars whose software could cause them to stop suddenly. Software bugs

have caused accidents, such as when a 2005 Toyota Camry accelerated through an intersection, killing one passenger and injuring the other [1].

In addition to the these unintentional errors and Volkswagen's deceitful coding, software engineers have demonstrated that the software code is vulnerable to attack by hackers. In July 2015, *Wired* magazine reported that two security researchers were able to control the air conditioning, sound system, and windshield wipers of a 2014 Jeep Cherokee [2]. They were ultimately able to cut off the engine while a reporter for the magazine was driving the car on the highway. The security researchers accessed these systems through the Internet. They disclosed the vulnerabilities to GM about nine months before they shared their research at a Black Hat conference in Las Vegas. At the conference, they omitted details about how they rewrote the ECU's firmware.

The code for the electronic control unit is protected under the Digital Millennium Copyright Act (DMCA), an intellectual property law, meaning that only the auto manufacturers have and can grant access to the code (such as to dealers to repair the cars). Some individuals and groups argue that car owners and independent security researchers should have access to the code. The Electronic Frontier Foundation has filed a request with the Librarian of Congress, who works with the Copyright Office, to "allow vehicle owners to repair, study, and tinker with their own vehicles" [3]. The Electronic Frontier Foundation has also requested an exemption to allow independent security researchers access to the code [4]. Car manufacturers oppose the requests [5].

Your Task

Decide whether security researchers and/or the general pubic should be allowed to study and modify automobile software code. Use the ethical tests (available on Blackboard) and/or an appropriate code of ethics to analyze and justify your position. Which principles of ethics do you base your decision on? You should take into consideration the interests of the various stakeholders, but you should still arrive at an ethical decision. If the results of the ethical tests indicate contradictory conclusions, you will have to decide which test should take precedent.
Be creative in solving this dilemma. Is there a solution that could satisfy all or most of the stakeholders in an ethical way?

The Stakeholders

- Automobile manufacturers
- The EPA and NTHSA
- The Electronic Frontier Foundation, a nonprofit watchdog that works "to ensure that rights and freedoms are enhanced and protected as our use of technology grows" [6]
- IEEE, a large and influential professional association for electrical and electronics engineers
- Car owners [and the general public]

References

[1] D. Gelles, H. Tabuchi, M. Dolan, (26 Sep. 2015), "Complex car software becomes the weak link," New York Times. Online: Available: http://www.nytimes.com/2015/09/27/business/complex-car-software-becomes-the-weak-spot-under-the-hood.html

[2] A. Greenberg, (21 Jul. 2015), "Hackers remotely kill a Jeep on the highway—with me in it," Wired, [Online] Available: http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

[3] K. Walsh, (4 Nov. 2014) "EFF to Librarian of Congress: Let car owners look under the hood," Electronic Frontier Foundation [Online] Available: https://www.eff.org/deeplinks/2014/11/eff-librarian-congress-let-car-owners-look-under-hood

[4] K. Walsh, (21 Jul. 2015), "Jeep hack shows why the DMCA must get out of the way of vehicle security research," Electronic Frontier Foundation [Online] Available: https://www.eff.org/deeplinks/2015/07/jeep-hack-shows-why-dmca-must-get-out-way-vehicle-security-research

[5] D. Kravets, (30 Sep. 2015), "Should research on vehicle software be hidden from the public?" ArsTechnica, [Online] Available: http://arstechnica.com/tech-policy/2015/09/should-private-research-on-vehicle-software-be-hidden-from-the-public/

[6] Electronic Frontier Foundation, (Accessed 4 Oct. 2015), "About us," [Online] Available: https://www.eff.org/about