JESA

The USENIX Journal of Education in System Administration

Volume 1, Number 1 • September 2015

usenix
THE ADVANCED
COMPUTING SYSTEMS
ASSOCIATION

JESA

The USENIX Journal of Education in System Administration

Volume 1, Number 1 • September 2015

In This Issue

ISBN 978-1-931971-263   ISSN 2379-4798

usenix
THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

The USENIX Journal of Education in System Administration
Volume 1, Number 1 • September 2015
www.usenix.org/jesa/0101

FOREWORD

JESA: A Milestone on an Interesting Journey

I remember seeing an actual milestone for the first time as a child, when walking along an old roman road in Germany. I didn't pay much attention to it. In fact, the whole village was called "Milestone," or Merkstein, so the fact that this was the stone that marked a certain distance to some city (not Rome) didn't convey the same meaning as it would to a traveler in olden times.

Now I look upon a new milestone which means a lot more, and that has just been placed. USENIX has for over a decade supported those who teach system administration. During the annual USENIX LISA conferences, we were granted a small workshop that combined teachers and practitioners: the system administration education workshop. Personally, my first workshop was in December of 2003. This was before the digital explosion that was to come with the likes of Google and Facebook. Virtualization was just about to burst on the scene, and cloud computing would follow half a decade later. In IT timekeeping, this was an age ago. Over the years, the group continued to show up and we heard of educational programs appearing. As IT exploded, we tried to keep up with updating our courses and programs, incorporating new languages, technologies and practices. We watched in fascination as the LISA community discussed the newfound place, role, and identity of their profession in the light of the IT revolution.

In 2012, I was approached to organize the first USENIX Summit for Educators in System Administration (SESA). This was a major step for our group. USENIX recognized the importance for the profession to have good and coordinated educational programs around the world and was prepared to back our group further. The first summit was in 2013, and next in 2014. These summits had a great effect on expanding the community and the recognition we got from the industry. Now, we witness our next and latest milestone: our first own-written body of knowledge.

It is our hope that through annual SESA events and *JESA* releases, the community, education, and identity of the system administration profession will continue to grow. SESA and *JESA* should not be the educators' voice, where educators write about education to other educators. It should continue to be the forum that is owned by the community for the exchange of ideas and experience. Our community is strong because it contains members of the industry as well as academia. We will continue to seek contributions about courses, programs, and pedagogical work as well as new trends and practices that happen locally and how the identity of the profession changes.

This first issue contains papers from educators and also a column from a member of the industry by the venerable textbook author, speaker, and former LISA Chair Tom Limoncelli. We consider this industry relationship to be extremely important as they are the profession we try so hard to prepare our students for. They also represent how it all started, with us under the wings of the premier conference of system administration.

We hope you will enjoy the first issue of *JESA*.

Kyrre Begnum, *JESA* Co-Editor-in-Chief

The USENIX Journal of Education in System Administration
Volume 1, Number 1 • September 2015
www.usenix.org/jesa/0101

INDUSTRY'S VOICE

An Open Letter to *JESA*

By Thomas A. Limoncelli, *StackOverflow.com*

I challenge *JESA* to create the next generation of system administrators (SAs) who can meet a wide range of ever-changing duties and responsibilities. As I will show, this is necessary for society as a whole to continue to grow and survive.

The Forest

Two people attempt to walk across a large forest.

The first person has no maps, no training, no tools or equipment. However, this person is very smart. Very, very smart. It will be difficult to make it to their destination. They will need to quickly figure out how to find food and water or they will starve. They will need to navigate based on educated guesses and not be deterred by missteps and dead-ends. Learning how to defend oneself against bears by trial and error seems like a losing proposition, but I suppose it can be done.

The second person has a recent map, training in outdoor survival skills, a proper compass, sleeping equipment, and a backpack. They know what plants are edible and which aren't. They know how to find safe drinking water. Recent plant growth has made their map slightly outdated but it is sufficient for their needs. Their training has prepared them to avoid bears and, more importantly, how not to attract them in the first place.

The first person's journey was high-stress, high-risk, and difficult work. When they were done they said, "I'm glad I made it!"

The second person completed their journey and said, "I'm glad my teachers prepared me so well!"

Society Depends on System Administration

We often lose sight of how important teaching system administration is. It isn't just important, it is scary-important.

When old systems are "computerized" this means "they now depend on sysadmins."

We are living in an era where society has become dependent on computers to sustain itself. It used to be that if the computer was down, we would temporarily "switch back to paper'' until the computer came back up.

We can't do that anymore. It isn't that IT is a part of how food gets from the farm to our plate, but rather that we, as a society, no longer know how to provide food without IT. The supply chain from fertilizer to supermarket is planned, executed and monitored at such a scale that it cannot be done manually. We literally do not know how to feed a city without IT. Medicine isn't just billed and administered with the assistance of IT, we literally can't provide medical services without IT anymore. Our entertainment, education, government, and economics are unsustainable without IT.

Excellence in system administration is key to sustaining civilization as we know it.

The Sad State of System Administration

Sadly, we're collectively doing a lousy job at system administration.

Very few IT organizations know and follow best practices. Many IT organizations know that they exist but don't follow them for political, budgetary, or competency reason. The rest, and I fear they are the majority, don't even know that best practices exist.

Students are graduating from four-year programs without understanding the internals of systems, nor how to manage large complex systems as they exist in the real world. This would be like auto mechanics not being taught how an internal combustion engine works or doctors somehow graduating medical school without knowing that patients are alive between office visits.

In other words, the state of IT is horrible. It is so bad that our jobs are being legislated. You know you're doing a bad job when legislation is created to set absolute minimum standards and yet they seem like difficult burdens. Sarbanes-Oxley essentially says, "If you are going to be so unbelievably stupid as to do backups without testing them, create accounts without having a mechanism to make sure they are disabled when the employee leaves, and let developers have unrestricted raw access to live databases, then we're going to legislate how you have to do your job." HIPAA essentially says that our industry has proven itself too incompetent to be trusted with securing databases or WiFi networks in hospitals.

Even though these laws require such ludicrously minimal basic best practices, they strike fear in the heart of the average IT department. It reminds me of stories that surgeons initially balked at the idea that they should wash their hands before surgery. President James A. Garfield might have survived assassination in 1881 if doctors hadn't probed his wound with dirty, unsterilized fingers and instruments. If that story made you think "euuuu," then I hope you feel the same way about sites that don't encrypt credit card information. Security compliance is hygiene.

System Administration is Becoming DevOps

In the 20th century, the Lean Manufacturing movement transformed manufacturing. It applied scientific operations management to manufacturing and revolutionized it, making it more cost-effective, faster, and more profitable. You may have heard it called Supply Chain Management, or Just-In-Time Manufacturing, or The Toyota Production System. It went from being a rare thing to the only way manufacturing is done at scale.

DevOps is the application of Lean Manufacturing to information services, and it is having the same transformational effect. To a system administrator it may feel like a new thing, but to the large companies that employ them, it feels like IT has finally woken up. DevOps is moving from a rare thing to the only way IT services will be delivered.

As I write this, a major retailer is in the process of laying off IT workers, yet it has 60 openings for people with DevOps skills. Another major company has 500 open positions for DevOps-related positions. DevOps is the way forward for web companies, startups, enterprises, retail, and all industries.

DevOps takes the risky, stressful nature of IT and replaces it with consistent, reliable results. People that work in a DevOps environment have less stress, higher job satisfaction, and feel more motivated. Sysadmins that have moved into DevOps environments do not want to go back to the old ways. This is why Gene Kim famously wrote, "The opposite of DevOps is despair."

We must not just teach the bits and bytes of system administration. Students must learn the high-level strategies of DevOps and the best practices and tactics that implement it.

What We Need

We need a new generation of sysadmins that are prepared to face the challenges of today and the future. Sysadmins need:

- **Deep technical knowledge of systems:** How CPUs, memory, storage, networking, and crypto work.
- **Software development experience:** How to automate systems, create tools, and build robust applications in support of operations.
- **The scientific process as applied to IT:** Engineering new systems and debugging existing systems should be based on science and evidence, not folklore and myth.
- **Strategies over specific technologies:** Technology changes rapidly, but strategies are timeless.
- **DevOps and SRE principles:** End-to-end system analysis, rapid iterations, automation, introspection, and balancing the need for change and stability.

- **A sense of duty and ethics:** With great power comes great responsibility.
- **The ability to adopt new technology:** We should instill a love of new technology as well as the wisdom to avoid snake-oil.

A **deep technical knowledge** of systems is required to administer today's large, complex systems. One can not design a large system, evaluate vendor claims, or fix a performance issue without understanding how modern multi-core CPUs work, how packets route, and the proper use and limits of crypto.

System administrators must be fully capable **software developers**. System administrators need to transition from being the people that do work to the people that maintain the automation that does work. Don't be the auto manufacturer assembly-line worker, be the person that maintains the robotics that build cars. This kind of software development requires dual expertise in systems and software engineering.

System administrators must be able to create automation so as to work smarter, not harder. They must be able to solve problems by glueing systems together using APIs and protocols. They must be skilled at creating new tools on demand using scripting languages. They must be able to create applications and self-service portals that relieve them from being in the loop, thus creating applications that empower and enable customers to be self-sufficient.

System administrators must also have enough software development experience that they can converse with developers in their own language, reason effectively when submitting bugs, and collaboratively design systems. They must understand what a developer means when told an algorithm scales O(n log n), yet they must also be able to explain when a CPU's L1 cache size is far more important.

We must encourage the use of the **scientific process** in IT to fight the constant flood of misinformation, myth, and rumor. Science requires measurement, experimentation, and iteration. It saddens me to see large projects fail because nobody took the time to measure actual latency, properly estimate storage requirements, or validate assumptions via evidence gathered through prototypes. It worries me that "Googling an error message" has become a first line of defense instead of a last resort after rigorous scientific methods have been exhausted. I fear that every time we accept cosmic rays as the cause of failure we are digging ourselves deeper into a hole. I cringe when, as happened recently, a technician tells me they'll never buy a hard drive from a particular vendor because the one in his home PC failed. I asked him which brand guaranteed supernatural perfection. He didn't have an answer.

Students should **learn strategies and best practices**, not just solutions for specific technologies. Technology is constantly changing. By learning generalized strategies and best practices, we prepare students for a career of constant change. Learning your fourth language is easier than the previous three because you've started to observe the common patterns between them. Strategies and best practices are the patterns that help us do our jobs as new technologies appear and evolve. Universities should be forbidden from offering vendor certification classes. Passing them off as "system administration" is harmful, unethical, and verges on malpractice. Would you trust a medical school that taught how to prescribe only one manufacturer's medicines?

DevOps and SRE principles are, essentially, bringing all aspects of operations science to the practice of IT systems and services. They take an end-to-end view, encouraging vertical integration over silos, and process and people over technology. They enable confidence to make rapid change and to innovate.

Because society depends on system administrators, and because of the power we wield, students must learn a **sense of duty**. There is an inherent responsibility and therefore system of **ethics** that are part of the job. Whether our systems control nuclear weapons, hospital monitoring equipment, the WiFi at a coffee shop, or a question and answer forum for English language and usage, our services exist because people depend on them. We have a duty to serve these people. Because the job of a system administrator inherently requires privileged access to systems and information, we have a higher bar for ethics. Every action we take has an ethical component: from not revealing personal information discovered while debugging a user's mail account, to deciding where to store backup data, to deciding how far to go in reporting when we see our employer is breaking the law.

Lastly we must teach the ability to adopt new technology. We should instill a love of new technology. To appreciate the value of change and innovation. The grandeur of how optimizing one part creates a cascade of changes. The way that each order of magnitude of reduced cost or size creates a new era of applications previously unimagined. Likewise, students must understand the Technology Adoption Lifecycle. They should appreciate how and why organizations are early adopters, technology laggards, and everything in between.

At the graduate level, students should go beyond the best practices and gain a deep knowledge of how they work. For every area of practice, they should know who originally created that area of practice, what the current accepted practice is in that area, and what the current cutting edge of that practice is.

Is System Administration about Failing to Fail?

IT operations is often measured not by success, but by lack of failure. We don't notice a system that runs well, but a well-run system requires huge amounts of planning, skill, and effort.

I've been told that the most difficult part of being a salesperson is that you must hear many "nos" to get to a single "yes." Salespeople learn to not be discouraged by the "nos." They are what teaches them how to get to "yes."

System administration is not about failure. Students need to learn that system administration is about iterations and repetition. At the micro level, we try, try, try before we succeed. At the macro level, we learn to iterate quickly through many failures and learn our way to success.

The Challenges Ahead

I see two major challenges for solving the problems and fulfilling the requirements I've listed above: creating a curriculum, and finding people to teach it.

One of the biggest challenges of creating a curriculum is fitting it all in two or four years. Just defining the minimum set of courses, to me, seems impossible. I'm a "yes/and" person.

It is tempting to focus the curriculum on building ideal systems from the ground up. However, the typical sysadmin's job is the opposite: they jump into an existing system which others have built, then slowly evolve it over time. Only once in my 25 years in system administration have I had the opportunity to start from a green field.

I think it would be useful to start students by giving them a completely working service and teaching them to maintain it and evolve it over time. Maybe each student receives a cluster of virtual machines that implement a web server, database, and load balancer. They would also be provided with a wiki of operational procedures which they would have to maintain. Assignments would be graded on completeness plus an evaluation of their updates to the wiki. Every so often the instructor would switch who was maintaining which cluster. Important lessons would be learned from inheriting their fellow student's messy systems and incomplete wiki documentation.

Another challenge is finding people to teach system administration. I hope *JESA* will be a forum to discuss successful ways to train, recruit, and retain excellent instructors.

To teach system administration, you must be a system administrator. In order to be able to teach relevant information, you will need to be an active systems or network administrator. Do real projects on campus or in industry. This is the only way to provide up-to-date, relevant, and applicable information.

Salaries in industry drain good people from academia. We are eating the seed corn.

Call to Action

System administrators are the architects, engineers, and stewards of the interface between the digital world and the real world. Creating the next generation of system administrators is a moral imperative. Society depends on it.

I am excited to see USENIX creating the *Journal of Education in System Administration (JESA)*. USENIX is in the unique position of being the center of system administration innovation for three decades and still being on the cutting edge of systems, security, and network research.

I challenge educators to create future generations of system administrators that can meet all of these challenges I've listed and new ones yet undiscovered.

Education is one of the top predictors of health and wealth. We are not in the business of making system administrators: we are in the business of producing a path to a healthy, wealthy, life for millions of people.

If your institution has a successful program, use *JESA* to give it the visibility it deserves and let others learn from your experience. When you try something new, share your experiences, both positive and negative. Use this forum to propose new curriculum standards, ontologies, teaching methods, and formats. Invent. Disrupt the old ways. Set a new standard and raise the bar. Have a sense of urgency: Passion builds momentum!

Solving all the problems I've listed above is a long, difficult journey. It is a journey without end. It is a journey whose importance cannot be understated. Society depends on system administrators in ways that even we ourselves barely appreciate. This dependence will only grow over time. The forest is deep and wide and full of unexpected challenges, beautiful vistas, and a lot of scary bears.

An old African proverb states: "If you want to travel fast, travel alone. If you want to travel far, travel together."

Now we can travel this forest together.

Thomas A. Limoncelli is a site reliability engineer at StackOverflow.com and the co-author of textbooks such as The Practice of System and Network Administration *and* Volume 2: The Practice of Cloud System Administration. *He blogs at everythingsysadmin.com and tweets @YesThatTom.*

The author wishes to thank the following for their feedback and review of drafts: Kyle Brandt, Katherine Daniels, Jason Punyon, Bart Silverstrim, and Sam Torno.

The USENIX Journal of Education in System Administration
Volume 1, Number 1 • September 2015
www.usenix.org/jesa/0101

Embedding Ethics in System Administration Education

Jeroen van der Ham, National Cyber Security Centre-NL, University of Amsterdam

Ethics is an important part of education in system administration. It is a hard subject to teach to science students, yet it is a pervasive issue in system administration and especially security research. Many student research project will touch on the security of users, their private data, or can even have implications for physical security.

In this article we demonstrate our approach for teaching and evaluating ethics. We start with regular lectures in ethics, but follow up with practical training by forcing the students to write ethical consideration paragraph in all of their project proposals. These proposals are then evaluated by an ethics committee for this educational programme. The ethics advisor is involved in supervising the project depending on the outcome of the ethical evaluation.

In this paper we also discuss several proposals that were submitted to the ethics committee. We discuss the deliberations, the eventual categorisation of the project, but also the outcomes of the projects. With these case descriptions we would like to improve the discussion on ethical aspects of system administration and security research.

1. INTRODUCTION

Computer scientists and engineers have long felt that ethics is an important part of their job: communities such as USENIX [Committee 2003], and the ACM [council 1992] have published codes of ethics. With the increasing popularity and dependency on the Internet, so too did interest in ethics in security research increase [Bailey and Kenneally 2014b][Bailey and Kenneally 2014a].

The System and Network Engineering Master education at the University of Amsterdam is a one year programme. In this limited time, students learn the theoretical underpinnings of networking, security, forensics and other related topics. The programme is designed so that students receive classical lectures in the morning. The rest of the day also consists of lab exercises: students must run their own servers, and do exercises. Most of the 12 courses contain a small project that the students do in teams or alone. Many of these projects are related to security or possibly sensitive data.

The term *ethics* in the context of this paper is restricted to the context of computer and information ethics [Bynum 2014]. In this paper we focus especially on the professional responsibility that is expected of system administrators, as encoded by the different codes of ethics in this field. In the field there is consensus that it should have some part in the curriculum, so this is often evaluated by the accreditation committee.

Ethics as part of the curriculum in the System and Network Engineering Master programme was limited to one or two lectures. That is, until 2014 when we started with a new approach. In addition to the lectures on ethics, we force the students to apply their knowledge on ethics. We require students to include an ethical considerations paragraph in all of their project proposals. To support and review this part of the project, we have formed an ethics committee. Since the start of the ethics committee in 2014, we have reviewed over 150 project proposals for eight courses.

In this paper we would like to share our experiences in setting up an ethics committee, the procedures we defined, as well as some example cases of ethical review of student projects. From our experience and as also seen in literature, it is important to share experiences with others. We are sharing this experience so others can learn from our process and experiences, but also about our ethics considerations, so that eventually we may come to a more uniform view.

The rest of this paper is organised as follows: first discussing related work in section 2, then section 3 describes the organisation of the ethics committee. In section 4 we describe several cases as handled by our ethics committee, and finally in section 5 we give our conclusions and some possible future work.

2. RELATED WORK

There is some difference in approach between the US and the EU regarding review of ethics. Universities in the US tend to have an Institutional Review Board[Wikipedia 2015b], which reviews any research-project which have any kind of involvement from human-subjects. Universities in the EU tend to have an Ethics Committee[Wikipedia 2015a], which generally are focused on medical research, focussing on reviewing research-projects involving human-subjects for clinical trials. The concept of having ethics committees reviewing computer science research-projects is still very new both in the US and EU, yet IRBs in the US are already somewhat more generic and accepted in fields other than medical research, which is not the case in the EU.

The importance of teaching ethics to computer scientists has been identified as early as 1989 [Couger 1989], and has resurfaced later [Dodig-Crnkovic 2004]. Extensive teaching material exists expressly aimed at computer scientists [Baase 2012].

The ETHICOMP conference series has a track on teaching computer ethics since 1995, and many related papers have been presented there, both on experiences in teaching ethics, as well as possible new approaches to ethics. These have been aimed at general computer science curricula, whereas system administration, and especially computer security, requires a different approach.

An extensive ethical review of a research project measuring the effectiveness of the PirateBay blockade has been published by this author [van Wynsberghe and van der Ham 2015]. The nature of this particular research project warranted an extensive ethical review; Bittorrent clients were identified without their permission, in order to measure the impact and effectiveness of a particular website blockade. This research project was performed at the University of Amsterdam, but did not take place in the context of the education programme, and predated both the institutional as well as the Master ethics committees.

The aforementioned CREDS workshops [Bailey and Kenneally 2014b][Bailey and Kenneally 2014a] have discussed different approaches, including the initial development of a model for best-practices [Dietrich et al. 2014]. A followup paper to the two CREDS workshops [Kenneally 2015] provides a good outline of the problems security researchers face, and provides some guidelines on how to deal with them.

This paper does not present a new framework, analysis method, or specific teaching material for teaching and reviewing ethics. In this paper we extend these ideas with a practical and more pervasive application of teaching ethics as part of the complete curriculum, specifically targeted at system administrators and computer security experts. The guidelines and frameworks have been used as input for the way the ethics committee analyses the proposals submitted.

3. ETHICS COMMITTEE ORGANISATION

Ethics as part of the System and Network Engineering Master education has long been a subject that we struggled with. We have attempted to teach students about this during several lectures, where we were able to reach some students, but not all. This was observed by the project proposals that were later submitted by students, which often included unethical subjects or research methodologies. For many subjects in the curriculum we combine abstract and practical approaches to teaching, but we did not apply this to the subject of ethics. This insight, combined with remarks in the 2014 accreditation report and changes in the institutional position regarding ethics prompted us to rethink our approach to teaching ethics. The accreditation committee gave a positive rating to the education, but provided two minor points:

> As two minor points, the panel recommends to address the ethical, societal and social aspects more strongly and to give the business organization aspects a more prominent place. [NVAO 2014]

The change in the position of the university towards ethics actually came about by a student project in the SNE Master; some years ago, students examined the security of a Dutch banking app. In the course of their project they managed to find a vulnerability in the banking app, which made it possible to perform a man-in-the-middle attack. The findings were kept secret initially, and the management of the faculty was alerted, while initiating a responsible disclosure procedure to the bank. Eventually this procedure completed successfully, but along the way the management and legal department of the faculty realised that they were not prepared for this. Combined with other developments, the faculty formed an ethics committee for the computer science department in 2013, proposing a procedure for handling research proposals. Proposals could be submitted to the committee for review, and the committee would review projects within two weeks.

3.1. Education on Ethical Aspects

The SNE Master programme is an intensive one year program, which include several courses with (research) projects. A timeline of two weeks for approval causes a significant risk for delay in the programme, courses take roughly eight weeks, and projects are usually a subset of those eight weeks. Additionally, the workload of reviewing all student project proposals would be impractical for the institutional ethics committee, there are roughly 30 students in this programme each year, with two courses in each block of eight weeks, this means an average of 30 proposals to review every eight weeks, just for this Master programme.

This prompted us to start an ethics committee for the SNE Master programme. This ethics committee is comprised of three members: the programme director, the security track coordinator, and an ethics advisor. The education of the ethical aspects in the SNE Master is comprised of several parts:

- An introductory lecture, in which ethical theories are introduced, and the evaluation procedure is explained,
- In a course with a (research) project, the students must write a project plan, which must contain a section on ethical considerations,
- The teacher of the course performs a first review of the project proposals, and provides an initial categorisation,
- The SNE ethics committee reviews the categorisation, and creates a final categorisation (within three days after the teachers' review),
- Students receive extra guidance from the ethics advisor appropriate to the categorisation.

3.2. Reviewing Ethical Aspects

Project proposals are put into four different categories:

Green. There is no possibility of ethical issues in this project.
Examples are offline analysis of very specific tools, or projects where no possibility exist to access sensitive data of third parties.

Yellow. There is a small possibility of ethical issues.
Examples are offline analysis of tools or operating systems, where an issue may allow others to gain access to sensitive data.

Orange. There is a real possibility of ethical issues.
Examples are research using personally identifiable information, obtained with prior permission, or sandbox analysis of important secure applications.

Red. There are clear ethical issues in this research.
Examples are research where (for whatever reason) personally identifiable information is obtained without prior permissions, or online analysis of applications involving third parties.

Once the proposals are categorised and approved, the students start with their projects, and a summary of the projects and their categorisation is copied to the institutional ethics committee. The projects are supervised as normal by the teacher and the teaching assistants. Additionally, the students are supervised by the ethics advisor on ethical aspects of their project. This ethical supervision is increased with higher categorised projects. The students are also encouraged to come to the ethics advisor if they find any ethical issues during their research.

Projects that are categorised as *Red* are first discussed in the SNE ethics committee. These projects are normally denied. If the ethics committee sees strong educational or societal value, then the project can be submitted to the institutional ethics committee for approval. The project can only continue if the institutional ethics committee approves the project. Furthermore, should students disagree with the ethical categorisation of their project, they can also escalate the procedure to the institutional committee.

3.3. Summary of experiences

The ethics committee for the SNE Master education started in 2014, since then the categorisation has been applied to over 150 project proposals, in 8 different courses (including repeated instances of the same course). As mentioned before we feel it is important to share our deliberations on these cases. We feel publishing these deliberations can help start the discussion, so that we can come to a somewhat uniform view on where the boundaries are for ethically acceptable research in system administration.

The students are warned before and during the projects that any issues they may encounter should be brought to the attention of the teacher or ethics committee immediately. This includes new insights into their project which would impact the ethical assessment, but also vulnerabilities discovered. Vulnerabilities will always be kept secret while the project is ongoing. Once sufficient analysis on the vulnerability is completed, a responsible disclosure procedure will be started with the relevant vendor.

Any responsible disclosure procedure is always done in accordance to the guidelines set by the National Cyber Security Centre [Centre 2013]. Summarised this means that any vulnerabilities found will not be exploited any further than to prove their existence and severity, no personal information will be downloaded from the system, nor will a service be disabled. The procedure is always initiated by the university staff, for reasons of responsibility, but also for continuity; these procedures unfortunately take a long time, and may extend well beyond the graduation of the student. For parties that have not published a vulnerability disclosure policy, we contact the party with an initial email describing global details of the vulnerability. In addition, we explain that we act in accordance to the before mentioned Responsible Disclosure-guidelines, request a statement that the company will not prosecute, and also set an initial, negotiable deadline for addressing the vulnerability.

4. CASE DESCRIPTIONS

This section contains descriptions of several cases that have been categorised by the SNE Master ethics committee. The examples given below are a selection of projects from the courses Offensive Technologies and Research Projects[1]. These examples were selected for illustrative value and diversity, other courses follow the same requirements and approach. The full reports of these cases are also available[2].

As mentioned before our approach for evaluation ethical considerations in student projects is inspired by the framework as described in [Dietrich et al. 2014]. We have extended that framework beyond the considerations of data sharing. An important aspect is the possible discovery of vulnerabilities in products, and their impact towards different groups: the vendor, the users, and the general public. We attempt to identify and balance all these factors when categorising the student research projects.

In some of the projects, the ethics advisor has suggested changes in the way that experiments were conducted, or how data was gathered. The advisor and the teaching assistants try to keep an eye on the students and their actions during the project. Since this is a Master level education, we do expect a certain degree of independence and responsibility from the students. We also instruct the students that the ethics committee is there for their protection; if they act within the bounds as defined by the ethics committee and advisor, the university will try to protect them as much as possible.

4.1. Project: Tinder Stalking

[1] See https://www.os3.nl/2015-2016/info/curriculum

[2] See http://staff.fnwi.uva.nl/j.j.vanderham/cases/

4.1.1. Project Summary. Tinder is a popular dating app, which allows users to discover other nearby users. This process of course depends on providing location data, which historically has not been properly protected by Tinder [Veytsman 2014]. Tinder uses information from Facebook profiles, for example the pictures used in the matching process are taken from there, and users are provided with an indication of overlapping interests from their Facebook profiles. In this research project the students attempted to verify whether Tinder had successfully implemented additional measures to secure the location data, and wanted to investigate whether it was possible to link Tinder users back to their Facebook profile.

4.1.2. Ethical Analysis. There is a clear risk in this project for obtaining information without informed consent from the participants. Due to the nature of this service, it was not possible to perform this research offline. Another important consideration in evaluating this proposal was that Tinder had been warned about possible problems multiple times in the past, and each time attempted to solve the problem. Initial approval for this project was given with the restriction that experiments would only be performed with either test profiles or profiles of their classmates with informed consent. With that restriction, and because Tinder had been repeatedly warned about this issue, the project was classified as *Orange*.

The students were able to query the server for nearby users, and a limited list of candidates would be returned. The approach required repeated query-results with different parameters for the same identifier. During the execution of the project the students discovered that due to the popularity of Tinder, it was hard to work with a limited set of profiles.

After discussion with the ethics advisor the experimental design was changed so that only a limited set of information from the query-results was used. Instead of looking for a single profile identifier, the students stored the location data, combined with a hash of the account identifier. This made it possible to perform the experiment, while maintaining the anonymity of Tinder users. The data would also be destroyed at the end of the project.

4.1.3. Outcome. During the project the students discovered that they were able to track location of Tinder users. As mentioned above, the methodology was designed to prove this result while preserving the anonymity of Tinder users. The students also found that (manually) using Facebook Graph search they could discover the profile of Tinder users, with high probability.

The staff attempted to initiate a responsible disclosure procedure with Tinder early 2014 several times, through several different contacts. Eventually the initial message was acknowledged, promising a response. After three months of no response, the staff decided to publish the report. As far as is known, it is still possible to track Tinder users using this method.

4.2. Exploiting Wireless Networking Memory Cards

4.2.1. Project Summary. Wireless networking memory cards are like regular SD memory cards. They provide access to a camera to store pictures, but at the same time a small System-on-a-Chip provides wireless networking capabilities, over which the files on the storage medium can be accessed. This provides a way wirelessly transmit pictures from cameras that do not have wireless networking capabilities. The students in this research project tested the security of the wireless networking implementation.

4.2.2. Ethical Analysis. Before this project no similar research on the security of these kinds of memory cards had been performed. Due to the limited capabilities of the implementation there was a significant possibility that the students would find vulnerabilities in this implementation. If found, these vulnerabilities would have far reaching effects on the viability of the products. The project would be performed in a lab environment, on prepared cards, to minimise the impact of finding a possible vulnerabilities. This project was classified as *Yellow*

4.2.3. Outcome. The students were able to identify multiple vulnerabilities in the different memory cards. They were able to reverse engineer the key generation process, for both the network and the access restriction on the card. Responsible disclosure processes were started with the manufacturers, and both responded. One manufacturer researched the vulnerability, acknowledged it, while pointing out that there was a mitigation strategy available for users. They were considering improvements in future products. The other manufacturer acknowledged the report, but did not seem to escalate the issue beyond the support desk.

4.3. Drone Hijacking

4.3.1. Project Summary. Communication with flying drones is mostly over some form of wireless networking. Most commercially available drones use simple networking that is often not secured very strongly, because of limited processing power and cost issues. Drones used for law enforcement should however have secure wireless networking. In this project the students examined the wireless communication used in drones that are similar to the ones used by law enforcement.

4.3.2. Ethical Analysis. Disturbing traffic to drones, and especially to those used by law enforcement agencies can be dangerous. While the experiments and analysis would be done completely offline, there is a possibility that vulnerabilities would be discovered during the project. Vulnerabilities in these products would have a severe impact, especially if they would become public. The project was conducted in cooperation with a drone manufacturer. This project was classified as *Orange*

4.3.3. Outcome. The students were able to record and analyse the communication between the drone and the remote. The video feed on the drone was actually transmitted over wifi, with weak security and default password. The control of the drone was sent over a separate channel. The students were not able to gain complete control over the drone, but they were able to use a replay attack on some commands, such as ‘start and lift off’, which starts the engine and lets the drone hover a few centimetres above the ground. These vulnerabilities were reported to the vendor, who acted on them quickly.

4.4. Peeling the Google Public DNS Onion

4.4.1. Project Summary. Since several years, Google provides a public DNS resolving service at the IP addresses `8.8.8.8` and `8.8.4.4`. Google published [Google 2015] that these resolvers are located in over a dozen different countries around the globe, but no information on the inner working of the resolving system is known. In this project the students tried to investigate the inner working of this service, and especially the cache distribution and the different cache levels within the service. The RIPE Atlas [RIPE NCC 2014] platform was used as a measurement platform. This allowed the students to use many different probes to submit queries to the Google DNS service in a geographically distributed manner.

4.4.2. Ethical Analysis. The project aims to analyse the public DNS service using public information, and regular queries to the Google DNS service. The number of queries was considered briefly, but this was quickly deemed negligible relative to the normal number of queries that are submitted to this service. The RIPE Atlas measurement platform uses credits for measurements, daily limits for each user, as well as global limits for each measurement target, to make sure that the measurement system is not abused. This project was classified as *Green*.

4.4.3. Outcome. The students used specifically created DNS zones and individually created measurements in the RIPE Atlas platform. During the project they discovered that the limits put in place in the Atlas measurement system restricted them from performing the research. After contact with the RIPE Atlas engineers, the usage limit for the students was increased temporarily for the duration of the project. Towards the end of the project, the students also contacted the Google DNS team to verify their findings. The team was cooperative, and confirmed some of their findings.

4.5. Evil SSD

4.5.1. Project Summary. The NSA had a programme called IRATEMONK [Schneier 2014], which uses a hard drive firmware to gain persistent access to a target computer. The NSA apparently used this exploit already since 2008. In this project the students researched the feasibility of hiding malicious code in SSD hard drive firmware, as well as the possible capabilities of malicious code in that firmware. For this project the students would be provided with a new SSD on which they could use an open source firmware, that they could adapt to test the possibilities and capabilities.

4.5.2. Ethical Analysis. The materials used for this project were all provided especially for this project, with no personal data on them. The students would perform their experiments in a controlled environment, without any external users. Any possible results would be published publicly, they do not concern a specific manufacturer of SSDs. In the project the students would try to adapt an open-source firmware for SSD drives[at Sungkyunkwan University 2015]. This firmware is meant for an SSD development platform, can only be used on specific older SSD models, and does not provide complete functionality (i.e. modifications of the drive contents are lost after rebooting).

There was a serious possibility that students would find significant capabilities for backdoors or hiding data, which can be used for malicious activities, however, any vulnerability would only work on this specific firmware. The research project and the published results would help the security community gain more understanding of what is possible with these kinds of advanced implants. With all this in mind, the project was classified as *Green*.

4.5.3. Outcome. The students were able to successfully implement a backdoor in the open source firmware. This backdoor is capable of modifying data as it is retrieved from the drive, presenting for example additional password hashes, or replacing complete binaries. This firmware however can only be used in one (old) specific type of SSD, and even then is not able to perform persistent writes.

4.6. RFID Lock Security Assessment

4.6.1. Project Summary. In the course of the study, students become more observant of possible security problems. One student observed that in a particular student housing, RFID locks were used throughout the building: both for entry to the main building as well as the personal dorm rooms. Tenants receive a personal RFID card which allows them to access their personal dorm room. It was already known that these RFID cards were not well protected, as it easily possible to read and copy the RFID cards. The student also observed that the locks in this building could be updated wirelessly, for example for a new tenant of a dorm room. In this project the students proposed to research the security of these wireless updating mechanisms.

4.6.2. Ethical Analysis. In the first proposal the students would be attempting the assessment at the dorm, which would mean that they would perform the observations on a live system. They would be attempting to capture and possibly replay radio signals in this live environment, possibly interfering with the security of systems outside of their control. This was categorised as *Red* and denied by the ethics committee.

After discussion, the ethics committee proposed that the student contact the owner of the dorm, to seek permission. Another option was to contact the manufacturer of the RFID locks to obtain testing equipment, so that the assessment could be performed offline. The student received negative answers from both the owner, as well as the manufacturer of the RFID locks. This meant that the project could not be performed within the context of the educational programme.

5. CONCLUSION

Ethics is an important part of education in system administration. It is a hard subject to teach to students who are used to exact sciences, yet it touches almost everything that they do. In this paper we have described our new approach to teaching ethics in the context of the System and Network Engineering Master education of the University of Amsterdam.

As a first step we have introduced an ethics committee to review student project proposals on their ethical aspects. In the project proposal the student writes an ethical considerations paragraph describing the possible issues relevant to the proposed research. This is to facilitate the committee, but also to reenforce the learning and applicability of ethics to the students. The projects are then categorised according to the possible risks and ethical issues that the students may encounter during the execution of the project. The intensity of guidance by the ethics advisor that the students receive depends on this categorisation.

In the second part of this paper we have described six different cases that have been evaluated by the ethics committee, with outcomes in several different categories. These project descriptions illustrate that ethical issues in education of system administration are numerous and diverse. They can range from privacy of users of external services, products, or software, but even to physical security, with drone communication and RFID locking systems. With these case descriptions we hope to engage with other ethics committees and other researchers to discuss ethical aspects of security research.

The experiences in reporting vulnerabilities were varied. Any vulnerabilities found by students are always reported to organisations using responsible disclosure procedures. Some organisations have ignored the reports, sometimes not even acknowledging receiving the report. This unfortunately means that setting initial deadlines is necessary. Most organisations though are grateful for the reports, acknowledge the issues and either act on them, or found the issues not serious enough to warrant action.

We have not received negative responses on our reports, besides the organisation not granting permission. It should be noted that disclosure procedures often take a long time. With small issues and cooperative organisations, they can be resolved within a day, but other issues can drag on for much longer. In one case we cooperated with a company to fix a vulnerability in their application, and the whole procedure took several months before it was finally resolved and published.

The experiences so far with the ethics committee have been positive. With the previous approach on ethics using lectures, the subject remained abstract and distant, even when discussing concrete examples. With the new approach, writing the ethical considerations paragraph forces the students to think about the issues, which leads to more open discussions, both with the staff, but also between students. The subject now also lives on through the whole programme, instead of only concentrated in the lectures.

An important consequence from starting our ethics committee is that it has increased internal support for the security research that the students are doing. With the committee we demonstrate that we have identified sensitive issues in the students research projects, and are capable of monitoring them.

Interest in computer ethics has increased, in society due to the pervasiveness of computers, and the practices of businesses and intelligence services. The field itself has also realised that ethics is an important issue to keep in mind, as big data experiments give rise to surprising privacy invasions, and vulnerabilities in software can have a serious impact on people or even society. The field of computer security should consider self-imposed regulation for reviewing ethics, before a more strict regulation is imposed by outsiders.

5.1. Future Work

With this paper we have demonstrated how we approach ethical analysis of student research projects. We are interested in hearing how other educational programmes in this field are handling similar issues. The case descriptions can hopefully help with a constructive debate, and perhaps as training material for similar committees.

For the analyses of the projects we roughly follow the general framework [Dietrich et al. 2014], in the future we would like to further fine-tune this framework to fit with the educational context of this field.

ACKNOWLEDGMENTS

The author would like to thank Jaap van Ginkel and Karst Koymans, the other members of the Ethics Committee, for their support and constructive discussion on these and other cases.

REFERENCES

Computer Systems Laboratory at Sungkyunkwan University. 2015. The OpenSSD Project. (2015). Retrieved 9 Septemeber 2015 from http://www.openssd-project.org/wiki/The_OpenSSD_Project

S Baase. 2012. *A Gift of Fire: Social, Legal, and Ethical Issues for Computers and Internet*. Prentice Hall, Upper Saddle River, New Jersey.

M. Bailey and E. Kenneally. 2014a. Cyber-security Research Ethics Dialogue & Strategy (CREDS) Workshop, CREDS II - The Sequel . (2014). Retrieved 28 July 2015 from http://www.caida.org/workshops/creds/1405/

M. Bailey and E. Kenneally. 2014b. Cyber-security Research Ethics Dialogue & Strategy (CREDS) Workshop Report. *ACM SIGCOMM Computer Communication Review (CCR)* 44, 2 (Apr 2014), 76–79.

Terrell Bynum. 2014. Computer and Information Ethics. In *The Stanford Encyclopedia of Philosophy* (winter 2014 ed.), Edward N. Zalta (Ed.).

National Cyber Security Centre. 2013. Responsible Disclosure Guideline. (2013). Retrieved 29 July 2015 from https://www.ncsc.nl/english/current-topics/responsible-disclosure-guideline.html

SAGE Executive Committee. 2003. System Administrators' Code of Ethics. (2003). Retrieved 27 July 2015 from https://www.usenix.org/lisa/system-administrators-code-ethics

J Daniel Couger. 1989. Preparing IS students to deal with ethical issues. *Mis Quarterly* (1989), 211–218.

ACM council. 1992. ACM Code of Ethics and Professional Conduct. (1992). Retrieved 27 July 2015 from https://www.acm.org/about/code-of-ethics

Sven Dietrich, Jeroen Van Der Ham, Aiko Pras, Roland van Rijswijk Deij, Darren Shou, Anna Sperotto, Aimee Van Wynsberghe, and Lenore D Zuck. 2014. Ethics in data sharing: developing a model for best practice. In *2014 IEEE Security and Privacy Workshops (SPW)*. IEEE, 5–9.

Gordana Dodig-Crnkovic. 2004. On the importance of teaching professional ethics to computer science students. In *Computing and Philosophy Conference, E-CAP*. Citeseer.

Google. 2015. Google Developers: Public DNS. (2015). Retrieved 30 July 2015 from https://developers.google.com/speed/public-dns/

Erin Kenneally. 2015. How to throw the race to the bottom: revisiting signals for ethical and legal research using online data. *ACM SIGCAS Computers and Society* 45, 1 (2015), 4–10.

NVAO. 2014. Assessment report Master System and Network Engineering, University of Amsterdam. (2014). https://search.nvao.net/files/53da2b8c1ff3d_rapport%20UvA%20wo-ma%20System%20and%20Network%20Engineering.pdf

RIPE NCC. 2014. RIPE Atlas. (2014). Retrieved 30 July 2015 from https://atlas.ripe.net/

B. Schneier. 2014. IRATEMONK: NSA Exploit of the Day. (2014). Retrieved 31 July 2015 from https://www.schneier.com/blog/archives/2014/01/iratemonk_nsa_e.html

Aimee van Wynsberghe and Jeroen van der Ham. 2015. Ethical considerations of using information obtained from online file sharing sites. *Journal of Information, Communication and Ethics in Society* 13, 3/4 (2015), 256–267. DOI: http://dx.doi.org/10.1108/JICES-10-2014-0044

Max Veytsman. 2014. How I was able to track the location of any Tinder user. (2014). Retrieved 28 July 2015 from http://blog.includesecurity.com/2014/02/how-i-was-able-to-track-location-of-any.html

Wikipedia. 2015a. Ethics Committee. (2015). Retrieved 9 September 2015 from https://en.wikipedia.org/wiki/Ethics_committee_(European_Union)

Wikipedia. 2015b. Institutional Review Board. (2015). Retrieved 9 September 2015 from https://en.wikipedia.org/wiki/Institutional_review_board

The USENIX Journal of Education in System Administration
Volume 1, Number 1 • September 2015
www.usenix.org/jesa/0101

An investigation of learning outcomes for MSc programs in Network and System Administration

Kyrre Begnum, Oslo and Akershus University College of Applied Sciences
Charles Border, Rochester Institute of Technology
Niels Sijm, University of Amsterdam

What is the essence of a graduate-level system administration education? What skills and abilities of the candidate should educators focus on when developing a new program? This paper investigates the learning outcomes from three MSc graduate programs in network and system administration. We use a tournament-based game as a survey to establish a ranking of all the outcomes from the programs. Our results show a clear emphasis on security and the ability to create working solutions based on abstract descriptions.

Keywords

Network and system administration education, Learning outcomes, Graduate programs, Program development

1. INTRODUCTION

Most aspects of our personal life now have a digital dimension, manifested through a service that is managed by network and system administrators. Be it our finances, travels, medical data or just socializing with our friends and relations. The increase in data-centers, cloud solutions and online e-commerce has opened up more jobs for sysadmins and a demand for professionals with expertise in that field, making it a viable career path for candidates with a computer science degree. In addition, the scale of the services and the complexity of managing them has brought it’s own academic field of research and inquiry within computer science. This means research positions and the prospect of PhD-level research within the field. Together, they drive the development of new educational programs that specialize in network and system administration.

There is no single path to becoming a sysadmin. Many have come from a different background and picked up the trade along the way. It is quite common to find that in a group of senior system administrators, only half would have a formal background in something that is related. This trend is changing, however. It is becoming increasingly common to find new employees from computer science graduates.

Over the last decade we have seen multiple educational programs appear, which are based on computer science and that offer a specialization in network and system administration. Students choosing such a program will be targeting the new and growing world of online services and large scale system administration as their career.

While most of the programs available are undergraduate (BSc), there are also graduate (MSc) programs. These offer a specialization for students with a degree in computer science and an opportunity to focus more directly on sysadmin-related topics. From tradition, they also focus more on the development of research skills within the field as students have to complete a thesis to earn their degree.

Developing a new program means writing a proposal for accreditation containing all aspects of the program, such as length, courses, curriculum, acceptance requirements and so on. The accreditation process is essentially the same in all countries as it requires the definition of learning outcomes (or student outcomes) for the student that captures the entire essence of the education [1; 5]. These outcomes inform the student what they can expect to have acquired by successfully completing the program. The courses will also have learning outcomes that reflect back and represent a portion of the program level outcomes. The course requirements and deliverables, such as tests, reports and presentations will assess whether the learning outcomes in fact have been met, validating the ”big picture” set out by the program-level outcomes. Program-level outcomes also communicate to the industry what topics they can expect the candidates to be proficient in.

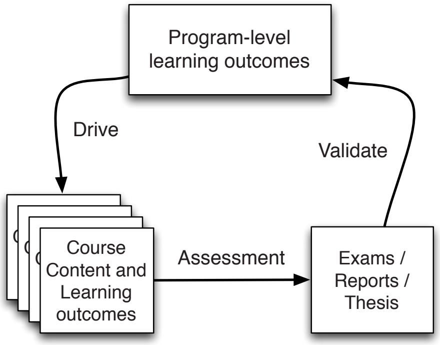

Program-level learning outcomes
Drive
Validate
Course Content and Learning outcomes
Assessment
Exams / Reports / Thesis

Fig. 1. The learning outcomes defined at the program level drive the learning outcomes, content and type of exercises etc. for the mandatory courses in the program.

In general, developing a program is a top-down process where you look at what overall learning outcomes the student should have and then what courses should cover these. For many established educations, such as teaching, nursing or engineering, the program level outcomes vary to a little degree. When developing a new program of the same topic, one therefore has a body of knowledge to lean on which has gone through accreditation. Today, in the case of a graduate program in system administration, the case is different. There is little comparison and standardization of graduate programs of this kind. This is unfortunate, as it is of little help for educators wanting to develop similar programs at their schools. Also, it makes it hard for the industry to engage in meaningful discussions about curriculum and courses, as they differ from program to program. Having a consensus of what topics would be considered common and constitute a universal kernel across the three programs would hopefully be a starting point for any new programs under development and also facilitate the discussion on whether the outcomes are properly aligned with the currents in the industry.

This article attempts to find what the core topics and goals of a graduate program in network and system administration should be by comparing and ranking the program-level learning outcomes based on three existing programs. The MSc programs at Rochester institute of Technology (Rochester, USA), University of Oslo and the Oslo and Akershus University College of Applied Sciences (Oslo, Norway) and the University of Amsterdam (Amsterdam, The Netherlands) are all examples of graduate programs that have started while the field was still in it's infancy. They were all developed relatively unbeknownst to each other and represent pioneer programs in their respective countries.

Our goal is to identify this kernel in the following steps:

(1) Compare the learning outcomes from all three programs and collect them into a set of outcomes
(2) Attempt a ranking of the outcomes based on the opinions of educators in the field
(3) Conduct a similar ranking based on student's opinions and compare the results with 2

By the end of this exercise, we hope to have found a consensus about what outcomes are considered most relevant to a graduate program in network and system administration.

The rest of the paper is organized as follows: Section 2 provides a brief overview of the three programs and present their learning outcomes. We will collect them into a set and discuss them form our perspective. In section 3 we will explain the methodology for our ranking. Section 4 will summarize the results and attempt to identify the core outcomes followed by a discussion and conclusion in Section 5.

2. BACKGROUND: PROGRAMS AT ROCHESTER, OSLO AND AMSTERDAM

In this section we provide a short overview of the respective programs and list their learning outcomes. As the reader will discover, the outcomes are expressed with similar phrasings and as a continuation of the sentence ”After successful completion, the candidate ...”. The authors have taken the liberty to modify selected outcomes in to singular form in order to improve the reading of them. Also, we have included a list of keywords, so that the reader more easily can extract the essence from each outcome. These keywords are our own attempt to classify the learning outcomes and do not represent any tradition in the development and presentation of learning outcomes.

2.1. Master in Network and System Administration at the University of Oslo and the Oslo and Akershus University College of Applied Sciences, Norway

This two-year program is a collaboration between the University of Oslo and the Oslo and Akershus University College of Applied Sciences. It is a MSc track within computer science that is available to graduates with a computer science or computer engineering degree. The first enrollment into the program was in 2003. Today about 25 students are accepted into the program each year. More information about the program can be found here: [10]

Compared to the other two programs, this has the longest list of learning outcomes. This is due to the local accreditation process in the Norwegian education system. The learning outcomes are to be described in three categories: Knowledge, Skill and General Competence. This increases the number of outcomes since a learning outcome might describe knowledge about a practice and then another would describe the skill of mastering said practice. The outcomes found in the general competence part, are often applicable for many programs, such as the ability to work independently and to complete a research project. In other accreditation bodies, such as ABET, these are embedded in the criteria themselves and do not have to be specified [1]. For the convenience of the reader, we have combined all three categories, since the other programs do not have this distinction.

2.1.1. Learning outcomes. After successful completion of the program, the candidate ...

(1) has thorough knowledge of the professions within network and system administration and their role in businesses, organizations and society
Keywords: Professional development
(2) has a thorough knowledge of the processes and methodologies applied by network and system administrators
Keywords: Processes
(3) has advanced knowledge of how network and system administration is applied at enterprise-scale organizations
Keywords: Processes, Scale, Enterprise
(4) can apply knowledge to new areas within the academic field of network and system administration
Keywords: Innovate, Science
(5) can analyze academic problems within the field of system administration based on its processes, tradition and role in society
Keywords: Analysis, Science
(6) can design and implement scalable and robust service architectures that represent modern and real-life scenarios
Keywords: Scale, Deploy, Industry-relevant
(7) can analyze existing theories, methods and interpretations in network and system administration and work independently on practical and theoretical problems
Keywords: Analysis, Top/Down, Independence
(8) can use relevant methods for research, academic and development work within the field of system administration in an independent manner
Keywords: Independence, Science
(9) can carry out independent research or development projects within the field of system administration under supervision and in accordance with applicable norms for research ethics
Keywords: Independence, Science, Ethics

(10) can apply methods and best practices in the field of system administration in order to evaluate and assess quality in the profession
Keywords: Analysis, Best-Practice

(11) can identify and communicate common facets and challenges within the field of system administration
Keywords:Analysis, Top/Down, Communicate

(12) can deploy, use and manage systems and services that in complexity and scale represent enterprise scenarios
Keywords: Deploy, Manage, Scale, Enterprise

(13) can design IT infrastructures that secure and ensure availability and quality of services and systems
Keywords: Infrastructure, Design, Quality of Service

(14) can analyze relevant academic, professional and research ethical problems in the field of network and system administration
Keywords: Analysis, Ethics

(15) can apply his/her knowledge and skills in new areas in order to carry out advanced assignments in the field of network and system administration
Keywords: Innovate, Learning

(16) can communicate extensive independent work and master the language and terminology of the academic field of network and system administration
Keywords:Communicate, Science

(17) can disseminate academic and professional issues, analyses and conclusions in the field of network and system administration to experts and non-experts alike
Keywords: Communicate, Science, Top/Down

(18) can contribute to new thinking and innovation processes
Keywords: Innovate

(19) has a professional attitude towards his/her field, including an awareness of ethical issues
Keywords: Ethics

2.2. Master in System and Network Engineering at the University of Amsterdam, The Netherlands

This program is offered at the University of Amsterdam and is available to students with a computer science or computer engineering background. Compared to the other two programs, this is a single-year track, however with a three-semester model. This makes it a rather intensive program, but students have reported a positive attitude towards this model, as it enables them to enter professional life earlier. The first year of enrollment was 2003 and today about 30 students are accepted each year. The contents and organization of the Amsterdam and Oslo program have perviously been discussed by Burgess and Koymans[3]. More information about the program can be found here: [11]

The number of learning outcomes are fewer compared with Oslo, but they contain many of the same keywords.

2.2.1. Learning outcomes. After successful completion of the program, the candidate ...

(1) has knowledge on an abstract level of the operation of computers and networks with respect to interfaces, protocols and software
Keywords: Top/Down, Network, Systems

(2) is able to translate abstract knowledge into concrete system and network configurations, independent of underlying vendor technology
Keywords: Top/Down, Network, Systems, Configure

(3) is able to acquire knowledge about innovative technologies and evaluate their potential
Keywords: Analysis, Learning

(4) is able to become acquainted with research methods in the domain within a short period of time and are able to apply these
Keywords: Learning, Science, Utilize
(5) is able to accommodate research innovations in an evolutionary way into existing systems
Keywords: Science, Utilize
(6) is familiar with the philosophy and practice of Open Technology and are able to evaluate its strength and possibilities in relation to proprietary technology
Keywords: Critical, Open Source
(7) is able to build innovative systems using Open Components
Keywords: Innovate, Configure, Deploy, Code
(8) is familiar with the ethical and juridical aspects of their research
Keywords: Science, Ethics
(9) is able to recognize security aspects of systems on all levels and to take adequate measures to eliminate security problems where needed
Keywords:Security, Top/Down

2.3. Master in Networking and Systems Administration at Rochester Institute of Technology, USA

This two-year program is offered at Rochester Institute of Technology in USA which also offers an undergraduate program with the same topic. Compared to the two other programs, this offers distinct tracks for students to take based on their preference. They are named *knowledge domains*, which are Management, Professional and Research. In addition, students can chose between a project or thesis option where they either complete a technical project or a more scientifically oriented thesis. First enrollment into this program was in 2007 and 17 students are estimated to begin in 2015. More information about the program can be found here: [6]

2.3.1. Learning outcomes. After successful completion of the program, the candidate ...

(1) will be able to describe technologies emerging in the field of networking and system administration and their impact on large organizations
Keywords: Top/Down, Communicate
(2) will be able to be a key contributing member in the development, management, or research of the computing infrastructure of an enterprise
Keywords: Code, Manage, Science, Infrastructure, Enterprise
(3) will be able to describe and implement technologies important to the management and deployment of large scale computing environments
Keywords: Communicate, Deploy, Configure, Scale
(4) will be able to interface and communicate effectively at all levels of an organization
Keywords: Communicate, Top/Down, Organization
(5) will be able to design and write effective computer and network policies that meet the operational and business goals of their organizations
Keywords: Communicate, Policy, Organization
(6) will be prepared to participate effectively in research positions, leadership positions, or professional careers in computing in both private and public sectors, or alternatively, for admission to other academic programs
Keywords: Science, Professional development

2.4. Comparison of learning outcomes

As all programs are well-established and have passed local quality assurance processes, there are no inappropriate or irrelevant learning outcomes. Every outcome is clearly a part of what one would consider valuable knowledge and skills for a career in system administration. However, we are interested in the relationship between them. When developing a program from its outcomes, it is

Rank	Score	Respondent 1	Respondent 2	Respondent 3	Respondent 4	Final ranking based on average score	
1	4	C2	C1	C2	C3	C2	3.50
2	3	C1	C2	C3	C2	C3	2.75
3	2	C3	C3	C4	C1	C1	2.50
4	1	C4	C4	C1	C4	C4	1.25

Fig. 2. This illustration shows an example of the tournament game. Every respondent completes one tournament, ranking the contestants C1 to C4. Every rank position is awarded a score which is collected and averaged to calculate the final ranking.

challenging to balance the amount of time spent on each outcome. Which one is more important? Where should the focus be?

There seem to be some keywords that are emphasized more at different programs. For example, at Rochester, the ability to communicate, participate and be a productive member is visible. At the Amsterdam program, Open Technology, Open Components and vendor independence is mentioned, which is not present in the other two. The Oslo program contains outcomes that speak to the ability to work independently, ethically and with a professional attitude, which stem from the learning outcomes in the general competence category.

3. SINGLE-ELIMINATION TOURNAMENTS FOR IMPORTANCE RANKING

The predictive power of tournaments is widely studied. This paper does not intend to summarize the wealth of knowledge accumulated in the field of statistics and all applications of tournaments and their varying formats. A good investigation into the relationship of noise and the predictive power of elimination tournaments has been done by Ryvkin et. al. [7; 9; 8]

In order to provide a ranking of all the outcomes, a pseudo single-elimination tournament-style game was developed as a form of web-based survey. In the tournament, all learning outcomes are seeded into rounds of matches. A survey respondent will be faced with a series of single matches with the text ”Which of the two learning outcomes is the most important for a MSc programme in network and system administration?”. The respondent must decide which of the two presented outcomes is the most important by clicking on it. The designated important is considered a ”winner” in the match and moves on to meet another winner from the same tournament round. The whole survey resembles that of a tournament, where the final winner is the one that has consistently been deemed most important by the respondent. In addition, all ”losers” are not immediately eliminated, but have to compete against other losers and so on. An example of this is in the soccer world cup or the Olympics Games, where the 3rd place is determined by the two semi-final losers. By including this process for all the losers, the result is a complete ranking where all the initial contestants will end up with a position. For example, the losers of the first round of matches compete for the bottom half of the positions. At the end of a tournament, each position is awarded with a score, the highest ranking getting the highest score. Since each respondent completes one tournament, all the scores will be collected into a cumulative score where the learning outcomes with a consistent high ranking will end up with a high total score.

The above described ranking system was implemented as a web-based survey. Individual match data as well as scores of the tournaments were stored in a database for later analysis. For practical reasons, a seed setting was included, since these tournaments made most sense when the number of contestants were in the power of two (4,8,16 and so on). In the case where the number of contestants was not the power of two, one could pick a seed that would be close, like 8 in the case of 11 contestants. In that case, every tournament would start with a random 8 from the original 11. The downside of this is that one does not fully control the number of tournaments every contestant participates in.

4. LEARNING OUTCOME RANKING BASED ON EDUCATORS AND STUDENT OPINIONS

During the 2014 USENIX Summit for Educators in System administration (SESA14), the participants were asked to complete such a tournament each with the results collected for this study. The summit represents a unique venue for educators and industry representatives to discuss and present topics with regard to education of system administration. The audience at SESA 14 were selected as an expert group for our survey, representing an international community of educators and industry participants who all have an expressed interest in system administration education by virtue of being present at SESA 14. All the learning outcomes from the three programs amount to 34, so a seed of 32 was chosen. This means that every participant only had a tournament of 32 learning outcomes, which was a considerable task with many individual matches. 13 anonymous tournaments were completed, which is 81.25% of the people present at SESA during the exercise. A total of 977 individual matches were registered.

A similar exercise was conducted on a class of 2nd year master students at the program in Oslo. The difference from the SESA14 group was a much lower seed, only 16. The reasoning for this was that since the students were asked via email to complete a tournament each, it was more unlikely that they would complete all the matches and end up with incomplete results. A total of 16 tournaments were completed with a total of 458 individual matches.

4.1. Ranking Results from SESA14

In order to analyze the results we will attempt to identify clusters of topics or outcomes if outcomes of similar phrasings are located close to each other and with similar scores. The resulting list of clusters or topics would hopefully reveal a clearer image of how the topics rank relative to each other.

Table I shows the top 16 outcomes. After the outcome and it's keywords, we see the average score attained by this outcome.

On top is a learning outcome describing the ability to recognize security aspects on all levels and apply needed measures, is the only learning outcome that describes security. The maximum score is 32, so an average of 25.7 is considered to be high. This is echoed by the median of the tournament positions. As this outcome was part of 9 tournaments (last column), it ended up in the top 6 half of those times. A closer inspection shows that it's lowest position in a tournament was 14. This learning outcome is interesting as it is the only one with *security* as a keyword. Security is a major topic within the field of system and network administration and it's high score makes sense.

On second place is a learning outcome that describes the ability to go from an abstract design to a concrete implementation. It captures, besides security, a major part of system administration as it involves many skills, such as abstract reasoning, service design, deployment, installation and configuration. Furthermore, it involves the ability to fill in the details that are omitted by the abstract description, which requires broad knowledge about local policies and technologies. It is, in a sense, the part of the job that ”only the system administrator can do” and overlaps very little with other jobs. The outcome is also unique and is the only *Top/Down* item that describes a technical task, the other ones are mainly about analysis and communication. It is understandable that this item got a high score, quite comparable with the first position with an even better median but a marginally lower average.

On the positions 3 and 4 we see two learning outcomes that both contain the keyword *Scale* and *Enterprise*. They describe advanced knowledge and the deployment of enterprise-scale services. Two other learning outcome contain *Scale*, which are on position 7 and 8 with a slightly lower average score but similar median. These four can arguably be grouped together under an ”Enterprise-scale” umbrella.

The learning outcomes on positions 5,6,9 and 10 are differently phrased but all speak about non-technical skills, such as designing services, understanding and assessing processes and best-practice.

Table I. Ranking results from SESA14, position 1 - 16

Position	Learning Outcome	Keywords	Avg. Score	Position median	Win percentage	Tournaments played
1	is able to recognize security aspects of systems on all levels and to take adequate measures to eliminate security problems where needed	Security, Top/Down	25.7	6	60.4	9
2	is able to translate abstract knowledge into concrete system and network configurations, independent of underlying vendor technology	Top/Down, Network, Systems, Configure	25.5	5	62.5	11
3	can deploy, use and manage systems and services that in complexity and scale represent enterprise scenarios	Deploy, Manage, Scale, Enterprise	23.6	8	62.9	11
4	has advanced knowledge of how network and system administration is applied at enterprise-scale organizations	Processes, Scale, Enterprise	22.8	8	62.3	12
5	can design IT infrastructures that secure and ensure availability and quality of services and systems	Infrastructure, Design, Quality of Service	22.3	8	73	12
6	can apply methods and best practices in the field of system administration in order to evaluate and assess quality in the profession	Analysis, Best-Practice	21	9	66.1	11
7	can design and implement scalable and robust service architectures that represent modern and real-life scenarios	Scale, Deploy, Industry-relevant	20.9	7.5	66	10
8	will be able to describe and implement technologies important to the management and deployment of large scale computing environments	Communicate, Deploy, Configure, Scale	20.6	8	67.7	13
9	can apply knowledge to new areas within the academic field of network and system administration	Innovate, Science	20.5	11	61.8	11
10	has a thorough knowledge of the processes and methodologies applied by network and system administrators	Processes	20.2	10	60.7	11
11	can apply his/her knowledge and skills in new areas in order to carry out advanced assignments in the field of network and system administration	Innovate, Learning	19.8	9	64.6	13
12	is able to acquire knowledge about innovative technologies and evaluate their potential	Analysis, Learning	19.8	11	54.9	9
13	can analyze existing theories, methods and interpretations in network and system administration and work independently on practical and theoretical problems	Analysis, Top/Down, Independence	19.7	11	59.2	9
14	will be prepared to participate effectively in research positions, leadership positions, or professional careers in computing in both private and public sectors, or alternatively, for admission to other academic programs	Science, Professional development	17.8	14	51.7	11
15	can use relevant methods for research, academic and development work within the field of system administration in an independent manner	Independence, Science	16.1	15	48.3	12
16	will be able to be a key contributing member in the development, management, or research of the computing infrastructure of an enterprise	Code, Manage, Science, Infrastructure, Enterprise	15.6	17	52.3	13

One could loosely describe them as tasks related to technical work and not to communication skills, as *Communication* is not mentioned in them. We could translate their general essence into ”Processes” and ”Service Management”.

The following 8 outcomes (11 to 17) are not very specific to system administration, as it details general traits of a successful student: to work independently and also be a good team-member, to be a good learner and to master advanced problems and theories. There are mentions of career options as well as academic work. The medians here are sinking gradually and we are at the point that could be described as the middle of the pack. The outcomes mostly win more than they loose, and in singular cases they end up on top. For example, number 14 actually came 1st in one tournament. It is more difficult to group these learning outcomes into one category, but what they have in common would be a trait usually associated with someone who has experience from a graduate program, namely the increased independent work, more theoretical tasks and overall increased level of difficulty. We therefore chose the term ”Academic proficiency” as a generalization.

The results continue on Table II. Outcomes from number 18 to 23 are less specific towards system administration. They seem to end up with a lower score than the ones that point to specific problems in the field, such as enterprise-scale services or security. However, it is interesting that most of them are about communication and interfacing with the organization. This is what is commonly described as ”soft skills”, although the above group may arguably be called a soft skill too. It is interesting to note how communication seems to cluster together like in Table II. As pointed out earlier, the fact that they have a low score does not make them irrelevant in a program, but it is clear here that the more concrete the learning outcome to the core of system administration, the more likely it is to be identified as more important than to ”be able to interface and communicate effectively at all levels of an organization”.

In a broad sweep, we will group the remaining outcomes into ”Scientific work”. With very few exceptions, we find the keywords *Science*, *Analysis*, *Innovate* and *Ethics* here. This is where we find the ability to become acquainted with research methods, conduct research by trying out new ideas and innovate as well as communicating results to the public.

4.2. Ranking results for MSc students at Oslo

These results have generally less data due to the smaller seed for each tournament. Also, the maximum attainable score from a tournament was now 16, so the averages seem at first lower, although they should not be interpreted as such.

The results form Tables III and IV are not well aligned with the SESA14 results, but there are some similarities. By comparing the student rankings with those from SESA14, we discover that about 80% of the outcomes end up 8 positions or less apart in the two tables. The median distance between an outcome’s ranking in the two tables is 6. There are a few cases where the distance is large, for example number 20, which is the second ranked outcome in the SESA14 table. The details about this learning outcome in this exercise are that it has participated in only 7 tournaments and ended on the positions 3, 5 (twice), 7, 11, 13 and 14.

These results do not give themselves to clustering the same way, due to the fewer number of matches and the fact that most outcomes have participated in fewer tournaments with less then half of the other outcomes. However, we can recognize some of the trends from the topic ranking from before. We see, for example, that the top 5 learning outcomes all address deployment, configuration and management of systems and services on an enterprise scale. The bottom of the ranking also repeats the same keywords, such as *Ehtics* and *Science*. We interpret the results that they show the same start and end as the SESA14 data, but do not identify strong clusters in the middle.

4.3. Identifying the core role of the MSc graduate

The tournament exercise revealed clusters in the 34 learning outcomes in the SESA14 data. Based on our analysis the order of the topics is as follows:

(1) Security
(2) Translating abstract descriptions into actual implementations
(3) Enterprise-scale
(4) Service Management and Processes
(5) Academic Proficiency
(6) Communication skills
(7) Scientific work

Table II. Ranking results from SESA14, position 17 - 34

Position	**Learning Outcome**	**Keywords**	**Avg. Score**	**Position median**	**Win percentage**	**Tournaments played**
17	has knowledge on an abstract level of the operation of computers and networks with respect to interfaces, protocols and software	Top/Down, Network, Systems	15.5	18.5	52.9	10
18	will be able to interface and communicate effectively at all levels of an organization	Communicate, Top/Down, Organization	15.3	20	47.6	12
19	will be able to design and write effective computer and network policies that meet the operational and business goals of their organizations	Communicate, Policy, Organization	15.0	13	44.1	11
20	can disseminate academic and professional issues, analyses and conclusions in the field of network and system administration to experts and non-experts alike	Communicate, Policy, Organization	14.9	20	50	11
21	is familiar with the philosophy and practice of Open Technology and is able to evaluate its strength and possibilities in relation to proprietary technology	Critical, Open Source	14.8	15	39.7	11
22	will be able to describe technologies emerging in the field of networking and system administration and their impact on large organizations	Top/Down, Communicate	14.3	17	53.6	11
23	has thorough knowledge of the professions within network and system administration and their role in businesses, organizations and society	Professional development	13.7	16.5	39.6	10
24	has a professional attitude towards his/her field, including an awareness of ethical issues	Ethics	13.3	22	37.3	11
25	can identify and communicate common facets and challenges within the field of system administration	Analysis, Top/Down, Communicate	12.8	15	40.8	9
26	can carry out independent research or development projects within the field of system administration under supervision and in accordance with applicable norms for research ethics	Independence, Science, Ethics	11.8	21	45	11
27	can analyze academic problems within the field of system administration based on its processes, tradition and role in society	Analysis, Science	11.6	20	32.1	10
28	can analyze relevant academic, professional and research ethical problems in the field of network and system administration	Analysis, Ethics	11.5	19	39	11
29	can communicate extensive independent work and master the language and terminology of the academic field of network and system administration	Communicate, Science	11.3	21	44.6	11
30	can contribute to new thinking and innovation processes	Innovate	10.8	23	35	12
31	is able to become acquainted with research methods in the domain within a short period of time and are able to apply these	Learning, Science, Utilize	10.5	22	36.2	11
32	is able to build innovative systems using Open Components	Innovate, Configure, Deploy, Code	9.8	21.5	25.8	12
33	is familiar with the ethical and juridical aspects of their research	Science, Ethics	7.8	25	25.5	9
34	is able to accommodate research innovations in an evolutionary way into existing systems	Science, Utilize	6.7	27.5	33.3	10

Table III. Ranking results from students, position 1 - 16

Position	**Learning Outcome**	**Keywords**	**SESA14 pos.**	**Avg. Score**	**Position median**	**Win percentage**	**Tournaments played**
1	can design and implement scalable and robust service architectures that represent modern and real-life scenarios	Scale, Deploy, Industry-relevant	7	11.1	3	71	7
2	can deploy, use and manage systems and services that in complexity and scale represent enterprise scenarios	Deploy, Manage, Scale, Enterprise	3	11	3	75	3
3	has advanced knowledge of how network and system administration is applied at enterprise-scale organizations	Processes, Scale, Enterprise	4	10.8	4.5	69.2	6
4	will be able to describe and implement technologies important to the management and deployment of large scale computing environments	Communicate, Deploy, Configure, Scale	8	10.8	6	61.9	5
5	can design IT infrastructures that secure and ensure availability and quality of services and systems	Infrastructure, Design, Quality of Service	5	10.5	3.5	55.6	6
6	will be prepared to participate effectively in research positions, leadership positions, or professional careers in computing in both private and public sectors, or alternatively, for admission to other academic programs	Science, Professional development	14	10.5	4	72.2	4
7	can disseminate academic and professional issues, analyses and conclusions in the field of network and system administration to experts and non-experts alike	Communicate, Policy, Organization	20	10.2	5.5	62.1	6
8	is able to recognize security aspects of systems on all levels and to take adequate measures to eliminate security problems where needed	Security, Top/Down	1	9.9	4	50	7
9	can apply his/her knowledge and skills in new areas in order to carry out advanced assignments in the field of network and system administration	Innovate, Learning	11	9.6	6	52.4	5
10	will be able to be a key contributing member in the development, management, or research of the computing infrastructure of an enterprise	Code, Manage, Science, Infrastructure, Enterprise	16	9.5	7	64.4	11
11	has a professional attitude towards his/her field, including an awareness of ethical issues	Ethics	24	8.7	6	41.4	7
12	has thorough knowledge of the professions within network and system administration and their role in businesses, organizations and society	Professional development	23	8.7	7	54.8	7
13	can apply methods and best practices in the field of system administration in order to evaluate and assess quality in the profession	Analysis, Best-practice	6	8.7	8	48.3	7
14	can use relevant methods for research, academic and development work within the field of system administration in an independent manner	Independence, Science	15	8.6	7	60	5
15	will be able to design and write effective computer and network policies that meet the operational and business goals of their organizations	Communicate, Top/Down, Organization	19	8.6	8	54.3	7
16	can apply knowledge to new areas within the academic field of network and system administration	Innovate, Science	9	8.5	7.5	55.2	6

Table IV. Ranking results from students, position 17 - 34

Position	**Learning Outcome**	**Keywords**	**SESA14 pos.**	**Avg. Score**	**Position median**	**Win percentage**	**Tournaments played**
17	can identify and communicate common facets and challenges within the field of system administration	Analysis, Top/Down, Communicate	25	8	7	45.5	5
18	is familiar with the philosophy and practice of Open Technology and are able to evaluate its strength and possibilities in relation to proprietary technology	Critical, Open Source	21	8	8	42.9	5
19	will be able to interface and communicate effectively at all levels of an organization	Communicate, Top/Down, Organization	18	8	10	51.6	7
20	is able to translate abstract knowledge into concrete system and network configurations, independent of underlying vendor technology	Top/Down, Network, Systems, Configure	2	7.7	7	54.8	7
21	can analyze existing theories, methods and interpretations in network and system administration and work independently on practical and theoretical problems	Analyze, Top/Down, Independence	13	7.4	7	48.4	7
22	has a thorough knowledge of the processes and methodologies applied by network and system administrators	Processes	10	7.3	9	53.3	3
23	can analyze relevant academic, professional and research ethical problems in the field of network and system administration	Analysis, Science	28	7	8	41.9	7
24	is able to acquire knowledge about innovative technologies and evaluate their potential	Analysis, Learning	12	7	8.5	51.4	8
25	is able to build innovative systems using Open Components	Innovate, Configure, Deploy, Code	32	6.8	10	42.9	5
26	can contribute to new thinking and innovation processes	Innovate	30	6.1	12	35.9	9
27	is able to become acquainted with research methods in the domain within a short period of time and are able to apply these	Learning, Science, Utilize	31	6	10.5	38.5	6
28	is able to accommodate research innovations in an evolutionary way into existing systems	Science, Utilize	34	6	11	35.3	4
29	will be able to describe technologies emerging in the field of networking and system administration and their impact on large organizations	Top/Down, Communicate	22	5.8	11	52.6	4
30	can communicate extensive independent work and master the language and terminology of the academic field of network and system administration	Communicate, Science	29	5.4	12	39.6	11
31	can carry out independent research or development projects within the field of system administration under supervision and in accordance with applicable norms for research ethics	Independence, Science, Ethics	26	5.2	12	38.1	5
32	has knowledge on an abstract level of the operation of computers and networks with respect to interfaces, protocols and software	Top/Down, Network, Systems	17	4.6	16	37	5
33	can analyze academic problems within the field of system administration based on its processes, tradition and role in society	Analysis, Science	27	4.2	13	33.3	5
34	is familiar with the ethical and juridical aspects of their research	Science, Ethics	33	3.2	13.5	24	6

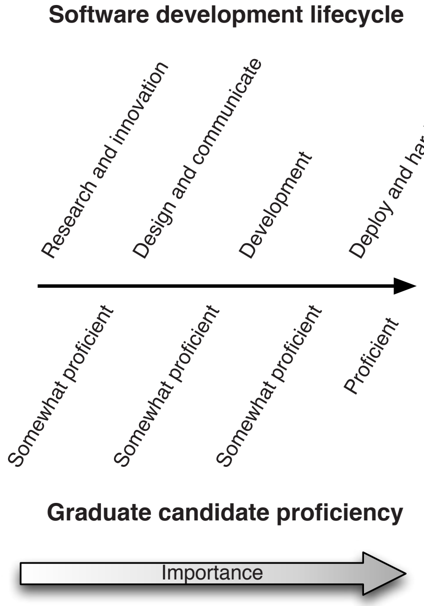


Fig. 3. A graduate level education in system administration is expected to provide some proficiency in many phases of a software development. During the final stages the expertise of the system administrator becomes more essential and is awarded more importance in the ranking.

It is interesting to see how the list goes from practical and concrete to the theoretical, underlining the focus of being a profession-oriented education. Security came on top because the only learning outcome with a security focus also got the highest average score. With more learning outcomes with a similar focus, we could have seen if the result would be consistent. In the case where there were several outcomes of the same topic we found that they ended up in proximity to each other.

Considering the ranking of the topics we can build a better understanding of where a candidate should be useful and expected to contribute. In a modern software lifecycle there are several phases where we find learning outcomes in our table that are applicable. There is research and innovation, design and communication, development and finally deployment and hardening. None of the learning outcomes can be omitted, so a certain proficiency in all fields can arguably be expected to some degree of a graduating candidate. But with other programs specializing in the earlier stages of the lifecycle system administrators would most likely only form a part of a team. In the deployment and stage, it is the expertise of the system administrator that is essential and requires them to be proficient to a higher degree on these areas than the preceding ones, See Figure 3.

Again, we need to stress that none of the outcomes and topics should be considered *unimportant*. The total spectrum of the learning outcomes showcase that the system administrator is expected to function as a team member in many areas.

5. DISCUSSION

During discussions at USENIX SESA or Sysadmin Education Workshop at the USENIX Large Installation System Administration conference, we find that many of the skills and knowledge presented in the learning outcomes resonate with what is considered important. However, there are two items which are missing: maintenance and troubleshooting. Undoubtedly, they make up a portion of a system administrators tasks, why are they not mentioned in the learning outcomes? In Figure 3, the software development lifecycle stops at deployment while every system administrator would argue that there is still a long path ahead after that with maintenance, monitoring and troubleshooting.

One explanation is that they might be considered ”helper subjects” that are not a topic in themselves but rather taught indirectly as part of lab work and exercises. However, there are some examples of maintenance tasks that in nature are so complex that certain strategies could be taught and discussed explicitly. One case in point is a rolling upgrade of many servers with a fail-over strategy. Another explanation for not focusing on troubleshooting and maintenance in the learning outcomes is that they could be considered something that needs to be taught at bachelor level, such as undergraduate elective system administration classes. However, this is not a given, as such courses are not available at all schools. It is not unlikely that a student completing one of the three programs mentioned here had no previous system administration courses as part of their undergraduate degree. Looking at undergraduate programs, such as the one at the University College of Gjøvik, we identify courses that address troubleshooting and maintenance to a larger degree[4]. Still, the authors believe that inclusion of these two topics should be discussed further in the future.

One could ask if a MSc graduate is a suitable candidate for a system administration position where the main tasks would be to maintain a small number of servers in a SMB context? Our results suggest otherwise. The fact that enterprise and scale were two important keywords, indicate that the SESA14 audience considered the candidates to be fit for larger and more complex environments where perhaps the approaches and solutions are not straight forward. If so, can a line be drawn between undergraduate system administration programs and graduate ones as to the scale and complexity of the infrastructure? Does the problem scope of a SMB infrastructure represent a good base for an undergraduate students, where certain skills and knowledge still needs to be acquired? Does teaching research and inquiry fit the more dynamic and complex problems of enterprise scale infrastructure? Our data do not answer this question, they merely help formulate it. Still, for us this question raises important points about how the profession and its practice can be properly aligned with the different program levels and will have to be investigated further.

Learning outcomes that spoke of programming scored low in importance. We believe this is because a background in computer science will mean that programming proficiency is present. The student is, in other words, expected to know how to write software and apply that knowledge in an operations setting, meaning scripting and automation. We find that automation combined with the ability to go from abstract to working solution is the basis for building modern platforms for services, such as continuous delivery frameworks and devops. The latter also includes insight into processes and service management, which is present in the learning outcomes as well. This is interesting, as a current trend in service architectures is that of the immutable infrastructure with clouds and containers[2], where software is not *maintained* in the typical sense, but simply re-deployed with a newer version.

The fact that the outcomes of similar topics ended up grouped together indicate that the ranking was rather consistent. This can at least be said for the top and bottom of the results. Also, the clear transition of topics from the concrete to the abstract can be interpreted that there is general consensus amongst the respondents that succeeding at the concrete and practical tasks is most important. SESA14 consisted of academics and members of the industry but the distinction was not recorded in the data. One could raise the question if there is even more consensus should we divide industry and academia into sub-groups. A future study with more targeted audiences would help clarify this. Still, we hope our findings and method can facilitate the development of future programs of the same kind by offering input into what outcomes should be included and were focus should be placed.

5.1. Future work

The authors plan to investigate the students opinions further, as there results did not yield as clear clusters as the SESA14 data. One approach would be to repeat the survey with the same seed (32) as for SESA14 and for students on all three programs. This would help us look for local differences in expectancies and provide more data for analysis. If the number of matches would make it impractical, a more condensed tournament consisting only of the keywords would also be possible.

6. CONCLUSION

This paper investigated the learning outcomes from three MSc programs in network and system administration. Our ranking experiment showed an order of importance that values security and the ability to deliver working solutions from abstract descriptions the highest. Theoretical work and ”soft skills” were lowest ranked. Our results suggest that there is consensus amongst experts as to what constitutes the essence of system administration.

REFERENCES

ABET.ORG. Us accreditation board for engineering and technology criteria for computing programs. http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-computing-programs-2015-2016/, August 2015. [Online; accessed August 2015].

BERNSTEIN, D. Containers and cloud: From lxc to docker to kubernetes. *IEEE Cloud Computing*, 3 (2014), 81–84.

BURGESS, M., AND KOYMANS, K. Master education programmes in network and system administration. In *LISA* (2007), pp. 215–229.

HIG.NO. Bsc in network and system administration at the university college of gjøvik. http://english.hig.no/study_programmes/bachelor/bdr, August 2015. [Online; accessed August 2015].

NOKUT.NO. Norwegian agency for quality assurance in education criteria for computing programs (norwegian). http://www.nokut.no/Documents/NOKUT/Artikkelbibliotek/Norsk_utdanning/Forskrifter_Kriterier_mm/NOKUTs_forskrift_vedlegg_2.pdf, August 2015. [Online; accessed August 2015].

RIT.EDU. Homepage of msc in network and system engineering at the rochester institute of technology. https://www.rit.edu/programs/networking-and-system-administration-ms, August 2015. [Online; accessed August 2015].

RYVKIN, D. The predictive power of noisy elimination tournaments. *CERGE-EI Working Paper*, 252 (2005).

RYVKIN, D. The selection efficiency of tournaments. *European Journal of Operational Research 206*, 3 (2010), 667–675.

RYVKIN, D., AND ORTMANN, A. The predictive power of three prominent tournament formats. *Management Science 54*, 3 (2008), 492–504.

UIO.NO. Homepage of msc in network and system administration at the university of oslo and oslo and akershus university college og applied sciences. http://www.uio.no/english/studies/programmes/nsa-master/, August 2015. [Online; accessed August 2015].

UVA.NL. Homepage of msc in network and system engineering at the university of amsterdam. http://gss.uva.nl/future-msc-students/information-sciences/content35/system-and-network-engineering.html, August 2015. [Online; accessed August 2015].

The USENIX Journal of Education in System Administration
Volume 1, Number 1 • September 2015
www.usenix.org/jesa/0101

The Development of a Computer & Network Security Education Interactive Gaming Architecture for High School Age Students

GUY HEMBROFF, Michigan Technological University
LUCAS HANSON, Michigan Technological University
TIM VANWAGNER, Michigan Technological University
SCOTT WAMBOLD, Michigan Technological University
XINLI WANG, Michigan Technological University

Computer and network security cases continue to rise each year, playing an important role within our society. With a growing job market in this field, there remains little formal education at the high school level to become familiar with this profession. We proposed to develop an interactive computer and network security game which differs from other security-based games previously created, as it does not focus only on computer science security. Our development focuses on a wide range of topics and layers of the OSI Model to offer computer and network security education critical in areas of network and system administration. We have created a storyline, in which each level relates to the story in sequence, creating an engaging story for the player. We also provide details how our gaming architecture is configured. Early results from players who have tested the game from a student and teacher perspective show encouraging results.

1. INTRODUCTION

Computer and network security education can be difficult due to the growing number of threats on computing devices or platforms, such as mobile technologies, routers, or databases. The wide array of applications, protocols, and procedures make computer and network security complex and difficult to fully understand. Universities have begun to offer courses and majors based around field, however most of the development surrounds computer science-oriented courses and often omits a considerable amount of computer and network security. Within most high schools, computer and network security education is often non-existent, and yet this age range is very impressionable and responsible for beginning to choose subjects of interest, leading to an eventual career path. Providing exposure, education, and ethical considerations surrounding the subject of computer and network security can prove beneficial in developing our world's next computer and network security specialists.

The motivation of this project focuses on providing a wide range of computer and network security to high school students through gaming, extending the research we conducted in developing information technology labs for universities with system and network administration programs [Wang et al 2013]. However, instead of offering lengthy documents about how to properly secure a network and computing environment, we provide a hands-on gaming environment to help immerse high school students into learning more about this field. This approach allows the project to encompass avenues of offense and the methods used to defend these types of attacks. The game is designed to provide a gateway for students to obtain knowledge of computer and network security in a fun and interactive environment.

Our motivation for this paper has two distinct purposes. First, we wish to describe our solution's mission to provide computer and network security education through gaming for high school students within a safe environment. Second, we provide detail in explaining the gaming architecture that was developed, providing details in its configuration. We chose to do this due to frequent complications associated with implementing a gaming architecture solution into your own network environment. We also thought the readers of the Journal of Education and System Administration would find the overall configuration details relevant due to its configuration content in system administration.

2. A RISE IN THE NEED OF COMPUTER AND NETWORK SECURITY PROFESSIONALS

A higher volume of cybercrime is conducted each year, while technology continues to evolve, placing more computing into our daily lives, giving humans more opportunities to store and access electronic data. According to the Bureau of Labor Statistics, "Since 2003, employment in the IT industry has grown by 37%." These careers ranged from programmers, systems design, computer facilities management, and other computer related services [Csorny 2013]. Cisco's Annual Security Report states that there will be a global shortage of over a million IT security professionals, stating: "Most people don't have the people or the systems to monitor their networks consistently" [Cisco Systems 2014]. With the expanding job market in computer and network related positions, the shortage for computer and network security personnel to secure these systems, and the complexities in providing this comprehensive security education, provide a compelling argument aimed to expose and educate high school students in this growing field by providing an engaging, safe, and fun-learning environment through gaming.

3. PREVIOUS WORK

Computer security gaming is not a new concept. Over the past ten years, there have been many developments in this field. While many past solutions differ from ours in terms of the audience they target or the approach taken, we have listed below those which have provided the most relevant content to our approach and the greatest influence to our project.

Research by Srikwan, uses cartoon comics to improve security awareness and understanding among typical Internet users, and targets similar age ranges as our research [Srikwan 2007]. The security topics covered include malware, spoofing, phishing, pharming and password safety. The cartoon comics provide simplified examples of common security threats/attacks, with the intent for the reader to understand the importance of internet security by drawing parallels between the example and the reader's life experiences. Although the visual comic provides an effective method for helping the reader understand why security awareness is important, it lacks the ability to go into detail in providing educational awareness for the user.

Development conducted by Jordan, Knapp, and Mitchell provide an interactive gaming concept, similar to ours, in training users in computer security [Jordan et al 2011]. Their creation of an interactive gaming concept to provide basic security training with surveys to evaluate the users' interests and opinions of the game was well-developed and helped to further our research efforts in this area. Our solution differs in providing a more comprehensive educational role within each module providing a continuing storyline to help keep the user engaged, and offer different security content.

Research conducted by ISECOM, a non-profit and open-source research group focuses primarily on security awareness [Hacker High School – ISECOM 2013]. The program uses a classroom lesson approach designed for teenagers as the main audience. There are currently nine subject areas with ten more in development. Topics range from learning basic commands in Windows and Linux, social engineering defenses, to using firewalls and Intrusion Detection Systems (IDS). Although the lessons include security-related exercises, this solution differs from ours in that it provides a much smaller amount of graphical content and does not offer detailed educational and module gaming material.

The group PicoCTF has developed a very interesting and comprehensive computer security gaming solution, which is accessible from a Web browser [Chapman et al 2014]. Although targeted age incorporates a wider range of students than ours, as it includes middle school age children, the security education gaming experience is rich in graphics and content, which is updated on an annual basis and offers open-source capabilities for development. Our security education differs in offering computer and network security content and education for system and network administration and does not focus solely on a computer science approach, as our solution includes modules associated with layers one, two, and three of the Open System Interconnection (OSI) model. The OSI model remains an integral reference and teaching tool into the organization of computer and network education.

4. METHODOLOGY

The project's goal is to create an engaging computer and network security game designed to broaden knowledge of basic to more advanced topics for users ages 14-18 years of age. A scoring server, to keep the scores of each player and allow them to compare and compete with other players is part of the architecture. Secondly, an intriguing story is important to capture the user's interest and keep them playing to find out what happens next. The story, as well as the game itself, is broken into modules, or levels. This helps to insure the player has smaller, manageable size modules to tackle, with each level providing a new computer and network security education topic.

We have aimed to improve upon existing tools already in use, and develop computer and network security education for the targeted audience. We plan to provide our gaming solution in a virtual environment, working with teachers of high schools to provide additional web-enabled tools and education for them to evaluate, improving the education of their students in this field.

5. GAME STORY

Our goal in writing the story and the layout of the game was to develop a linear path that would allow players to see how each scenario that applied in the real world. We wanted to show scenarios from each layer of the OSI model and present discussion on the legal and ethical implementations of each scenario, as well as give an overview of how organizations approach such a case. In the story, players take the role of a competitive programmer/hacker 'Oddball'. Your friend 'Shortstack' won the Pwn2Own contest with a Cross-Site Scripting attack against Amazon. Upon your arrival home, Shortstack is arrested as his program, 'pikpokit' was found to be in use. The FBI agent approaches you to be a consultant on the case. In the first level, players will learn a little bit about social engineering and communication as they talk with Shortstack. The next level has the player looking at Shortstack's laptop, trying to find proof of it being tampered with. The player will find a USB key, which contains a program they will have to interpret. The program is a keylogger, sending keystrokes to an IP that is scanned by the user in the next level using nmap. The networking levels use VMs to create safe yet realistic environments for the players to work in. Scanning the IP turns up a home Web server of a script kiddie. There the player will learn about social engineering and communication. The player will then discover that the script kiddie was just passing along the information to a larger cybercrime organization. The next hop shows Shortstack's data was sent to a Web server for a small store in northern Vermont. Here, the player will show how the server was hacked using SQL injection. They will then write access-control lists (ACLs) and configure to help prevent these kinds of attacks in the future, as well as look at phishing emails used by the crime syndicate to gain information about their target. These phishing emails point to another server where the data is being stored to distribute to 'pikpokit'. Using a man in the middle attack, the players will find this information. The player will then decrypt the data found using python to run the Blowfish and AES encryption algorithms. Seeing that they are compromised, the syndicate runs a DDoS attack against the FBI, which the player helps mitigate. The story ends with a final interrogation of an underboss in the syndicate, and has different endings based on the success of the player.

When choosing the security content of the game, we aimed at providing education around critical computer and network areas, along with considering the various layers of the OSI model. We choose to embed strong conceptual content over specific applications-type of computer security, such as Heartbleed, to ensure students would learn fundamentals, and not merely the latest attacks. Modules were also constructed to provide the users with combinations of communication, observation, and technical skills.

6. LIST OF MODULES

Modules were developed to provide a fun and interesting way to learn about computer and network security concepts through interactive demonstrations. The modules cover a wide variety of security topics coinciding with OSI model layers. Additional modules can be created and added to the game, permitting the game to evolve with computer and network security education. The only caveat depends on following the sequence for the existing storyline.

6.1. Module 1: Physical Security

This module simulates the "friend's" laptop that had been tampered with at the hackathon event. The player will look at various different images to view parts of the laptop, which contain clues to signify a malicious event, such as missing screws from the laptop chassis, or damaged physical ports on the device. The player to pick from certain options to identify information needed to understand how the code was removed from the laptop and used for malicious purposes. The interface is simple enough to see the images of the laptop before and after the tampering occurred. This allows the player to identify certain areas of the laptop that have these indications.

6.2. Module 2: Port Scanning

In this level, the player is asked to look at a machine discovered from the keylogger software on the friend's computer. The player will use Nmap, a piece of software designed to scan and interpret what ports are open on a computer. With this information, the player can determine what the machine is being used for and if there are any open ports used against the machine. This level is designed more around information gathering than it is about attacking or defending.

6.3. Module 3: Keylogger

In this module, the user is presented with a piece of code that was discovered on their friend's laptop. The goal of this level is for the player to determine what the code does and where it is sending the information. With help from the affiliated education section, the player will determine it is a key logger used to send information to a server controlled by the malicious cartel.

6.4. Module 4: Social Engineering

The social engineering module for the project changes focus from the technical aspects of computer security to the interpersonal relationships which take place within the environment of an established network. By definition, social engineering is the idea of a network's security being penetrated through human interaction and the manipulation of individuals. Although it is important to focus on securing a network through its perimeter, the idea of an individual within the company also poses a risk which often goes neglected. A study by Check Point Software Technologies, which included 833 IT professionals from around the world, discovered that among these professional companies, only 26% of participants actively train employees on the threat, 34% did not have any initiatives in place during the time of the study, and that 40% put the responsibility on the employee to read and understand their organization's overall security policy documents to prevent data loss, security attacks, and social engineering-based threats [Check Point Software Technologies LTD 2011]. This module is intended to enlighten the user of the effects of social engineering in its many forms, and often as a non-technical approach in computer and network security.

The primary focus of the module is to give the player an understanding of nonverbal communication to assess a situation. Using Ren'py, a distribution of the Python scripting language specifically designed around ease of storytelling and education, the user switches from being the player who is attempting to set their friend free, to an FBI agent during an interrogation with the initial suspect. The module is intended to be broken into two separate sections. The first occurring in the early stages of the game to familiarize the user to the concept of shared body tells among humans. It functions as a tutorial, providing information presented in flashbacks to the night of the initial crime based on given reactions to the accused during questioning. In the secondary part of the module, which occurs towards the end of the game, the user is able to use what they had gathered from the initial module and perform the interrogation with a suspect where no tutorial is provided.

Scoring on the combined modules is dictated towards an internal point system the scoring engine uses. Points are added together based on reactions and information the player receives from the accused, and from there, an ending is given based on the amount of points received. However, if the user returns with no further information, and the accused refuses to speak, a scripted video will appear as if a new interviewer were to go in with the accused and obtain the information themselves, providing a tutorial in how this technique works.

6.5. Module 5: Encryption

This Encryption is the process of scrambling data so that a user would need a key to properly view the data. This module focuses on the math behind two common symmetrical encryption algorithms, AES and Blowfish. Symmetrical algorithms use a single key to encrypt and decrypt information, and are usually done to protect files at rest. Users will be presented with unfinished scripts for AES and Blowfish implementations, where they must finish the lists of matrix transformations for two algorithms.

6.6. Module 6: DDoS Mitigation with SYN Flood

The DDoS mitigation module will educate users on how to mitigate a syn flood attack against a Web server. The player is given a computer on a public network trying to access a Web page that resides on one of the FBI's private networks. The player will enter this module as the attack has already begun and is tasked with needing to regain access to the Web server. The more time a player takes, the total amount of potential points of the level diminish. The Web page entails information for the player to submit, validating they have got the Web server back up and running. The player will be asked to do all of this by using SSH to access the NetBSD packet filter placed between the public and private networks which the Web server resides. On the NetBSD machine, only certain commands are permitted for the player to issue. These commands are already set up in a sudoer file to keep the player on track. For example, if the player issues a *tcpdump* command on either interface pcn1 or pcn2, it will display a large amount of syn's being sent to the server, with zero of the handshakes completing successfully. This will inform the user it is most likely a syn flood attack. The player must alter the rules of the packet filter and stop the attack. The rules to be used in the pf.conf file will be TCP SYN proxy and anti spoof. With the correct rules put in place, the player will be able to view the Web server's Web page from their computer and answer the question to validate they have completed this module successfully.

6.7. Module 7: SQL Injection

This level brings players to an offensive exercise, where the user is given a Web page for a small business, which has been configured with vulnerabilities, allowing the malicious cartel to access the small company's machine and use it as a data drop. The player is given the address to a Website of the vulnerable Web server and performs various SQL injection techniques to gain access to information stored in a SQL database. This information can then be used to log onto the server being used by the small business. This information will also give them root access to the machine, allowing them to locate the data linked to the malicious cartel and the next clue in tracking this group.

6.8. Module 8: Phishing

This level is designed with a list of emails. All of the emails are presented to the player in their source format. The player goes through these emails and is shown various forms that a phishing email can take. Most of the emails presented to the player are considered legitimate and are not Phishing emails. The player's task is to select the correct phishing emails to successfully complete the level and gain experience in detecting this type of malicious content. Each module we have developed offers a corresponding education section to help players learn about the respective subject areas and become knowledgeable enough to accomplish the level, regardless of prior education in this area.

6.9. Module 9: Man in the Middle

In this level, the player is attacking a server they believe has the *pickpocket* program, leading the user to the cartel. The player simulates a man in the middle attack using Kali Linux, while using specific tools to perform such an attack. The user utilizes Ettercap, which will permit them to locate active traffic on the network where the player will then poison this traffic link between two hosts and obtain the sensitive information as an acting host to each of the two original hosts. Using Wireshark, they can capture more detail about the traffic, such as the protocol being used and the ability to follow the TCP stream, finding the details of the information being sent over the link. With this module, it is important for the user to learn what types of attacks are common, and the methods that are performed, so they can learn how to prevent such an attack.

7. GAMING ARCHITECTURE

7.1. Virtual Machine Web Service Platform

To provide an interactive gaming solution, we created a Web-based virtual machine platform. The solution was developed to be hosted on open-source and provide a user-interface which connected to a backend of virtual machines, coordinating the security, scoring, and virtual machines needed within each module or level. The solution was also developed to be free, or relatively inexpensive, as this project was designed to keep costs at an affordable level. A total of five different services were compared.

The first alternative used VMware WSX to connect to the virtual machines remotely from a browser [VMware 2012]. This platform was very fast and provided a variety of options to turn off, pause, or administrate virtual machines remotely. This tool however, was not open source, and as a result could not be configured to the gaming structure's specifications. VMware VSphere Web client featured a rich set of options, including even more tools to administer the virtual machines [VMware 2013]. This connection, however, could not become Web-based and was not open source. PHPVirtualBox is an open source product and included administrative controls needed to administrate the servers, as well as protect the virtual machines and communication [PHPVirtual Box 2013]. However, this option struggled to integrate the environment and tools we had developed, and therefore was not considered a viable solution. The next option was a commercial solution utilizing RealVNC [RealVNC 2013]. This solution would cost approximately $900.00 for the license. The last option was the most optimal for our design. It also consisted of utilizing VNC, however, this solution incorporated VNC over a WebSocket, or NoVNC [Martin 2013].

NoVNC was an open source solution and also included a powerful pre-built library to incorporate many of the simple features we needed. The copy function also provided a means to pull data from the user's virtual machines within the game and send over WebSockets to the scoring server securely. The connection options also provided the debugging features we required. A wide array of protocols and security features are included with NoVNC to ensure the game is played properly and securely. Cookies are used to remember settings and restore defaults when connecting to the virtual machine's Web service. Secure WebSockets (wss://) are used to transfer information from the clients to the servers. This includes the remote frame buffer (rfb) information while controlling the virtual machines, as well as the data from the user to be scored for each module. The connection starts on the Website where the WebSocket is opened, and a connection using SSL (wss://) is created. The controlled virtual machines host a WebSocket proxy and receive the information from the wss://. This proxy forwards the stripped rfb protocol to the VNC server, which is hosted on the virtual machine controlled machine.

7.2. The Virtual Machines of the Gaming Architecture

The Multiple administrative servers exist on the architecture, outside of the virtual machines designated for each module. Scoring servers, Web servers, and MySQL servers are needed to ensure secure and robust transmission of data between the virtual machines. Figure 7.1 illustrates the computer and network gaming architecture.

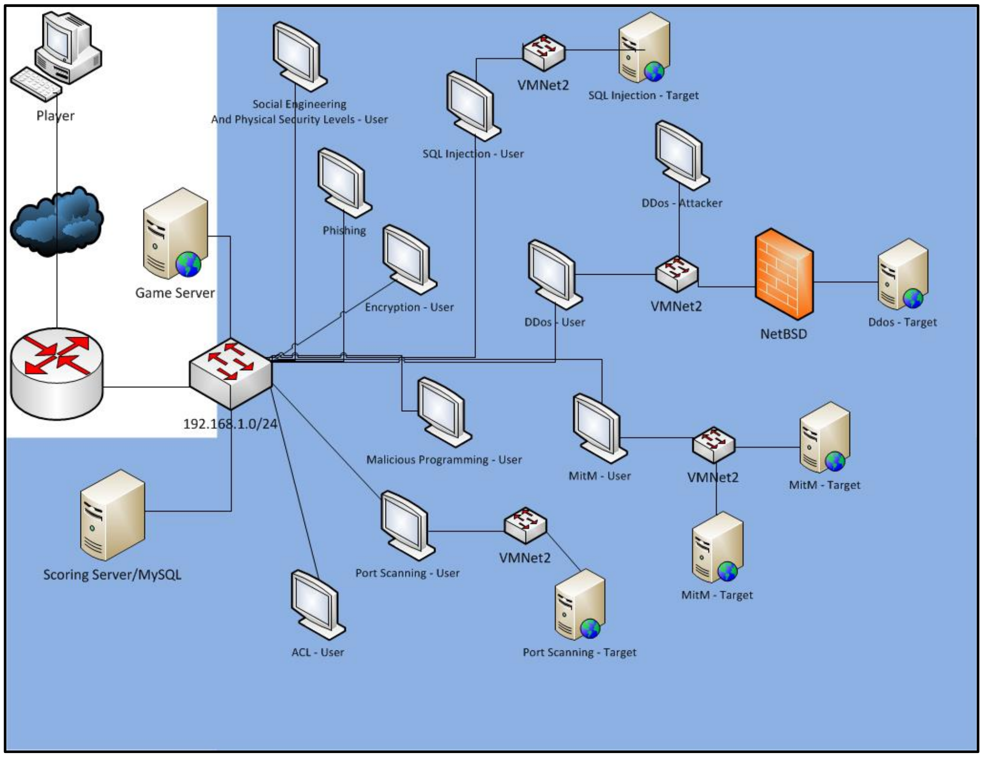


Figure 7.1. Computer & Network Gaming Architecture

Each separate module's virtual machines are Kali, Fedora, Windows, or NetBSD based. Each module's virtual machines require the Websockify proxy, as well as a VNC server running at all times, permitting access from Web clients to the virtual machines.

The Web server uses the solution created by Apache called Xampp on the machine facilitating the Website for clients to use. The Web server incorporates NoVNC as the virtual machine Web service platform, as well as a WebSocket client for communication to the scoring server and virtual machines. Help topics, Camtasia walkthroughs, network diagrams and technical help have also been added and hosted on the Web server and served by an "as requested" system.

7.3. VM Resource Management

The use of one or multiple VM's for each module presents potential resource allocation issues and can slow or halt the game sequence, which leaves users frustrated or disinterested in the game. Early in the development of the game, we experienced some levels having too many virtual machines for our physical computer to run efficiently, causing slow response times. To rectify this issue, we implemented the VMs for each level to start and stop on command. VMware workstation has a useful command called *vmrun*, which allowed the game's VMs to start, stop, and revert to snapshots using the command line. These commands were then added via a batch file, which we call PHP when machines needed to start, stop, or be in another state required for each level.

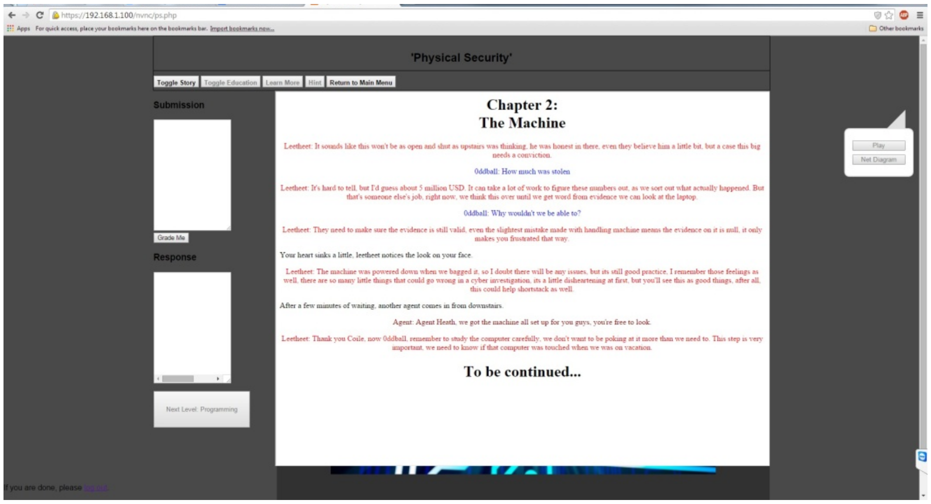
Each batch file starts off with stopping every VM that is part of the game's respective levels. This ensures that no other machines are turned on by accident, interfering with the game or consuming its resources. The batch file then runs the command *vmrun reverttosnapshot*, which will revert any machine for that level back to the state before a player had started to use it.

7.4. IP Address Schemes

The Connections to machines utilize the private space of 192.168.1.0/24 network. All VM's used by the player have a 192.168.1.x IP address on a network interface which has a bridged connection to the physical machines network. The VM's also have additional virtual private network space they require to perform the offense and defense exercises within each module. These networks range from 172.16.0.0/16 to 10.0.0.0/8.

7.5. Web Frontend

The Web frontend, created in PHP, is used as the Web interface for the players to interact with the game, providing paths to interact with the backend servers. If a user accesses the education portion of the module, they have an option to press the *Play* button on the right side of the screen and view the virtual machine(s) used for the specific level, to becoming accessible. Other buttons on the screen are intentionally disabled when they are not relevant and help to keep players on track. Figure 7.2 provides an example of the gaming Web frontend.


https://192.168.1.100/nvnc/ps.php
Apps For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Other bookmarks
'Physical Security'
Toggle Story
Toggle Education
Learn More
Hint
Return to Main Menu
Submission
Grade Me
Response
Next Level: Programming
Play
Net Diagram
If you are done, please log out.

Chapter 2:
The Machine

Leetheet: It sounds like this won't be as open and shut as upstairs was thinking. he was honest in there, even they believe him a little bit, but a case this big needs a conviction.

0ddball: How much was stolen

Leetheet: It's hard to tell, but I'd guess about 5 million USD. It can take a lot of work to figure these numbers out, as we sort out what actually happened. But that's someone else's job, right now, we think this over until we get word from evidence we can look at the laptop.

0ddball: Why wouldn't we be able to?

Leetheet: They need to make sure the evidence is still valid, even the slightest mistake made with handling machine means the evidence on it is null, it only makes you frustrated that way.

Your heart sinks a little, leetheet notices the look on your face.

Leetheet: The machine was powered down when we bagged it, so I doubt there will be any issues, but its still good practice, I remember those feelings as well, there are so many little things that could go wrong in a cyber investigation, its a little disheartening at first, but you'll see this as good things, after all, this could help shortstack as well.

After a few minutes of waiting, another agent comes in from downstairs.

Agent: Agent Heath, we got the machine all set up for you guys, you're free to look.

Leetheet: Thank you Coile, now 0ddball, remember to study the computer carefully, we don't want to be poking at it more than we need to. This step is very important, we need to know if that computer was touched when we was on vacation.

To be continued...

Figure 7.2. Gaming Frontend Web View Example

The main screen provides a window mirroring an instance of the chosen module's virtual machine. The virtual machine is selected from the right sidebar, with each button representing a different machine. When the button is clicked, the designated identifier is pulled from the utils.js JavaScript file and called into the Web interface's PHP file. Full control of the machine is allowed through this portal.

Other buttons, listed on top, such as the *Learn More* and *Hint* buttons, provide additional information specific to each. The *Learn More* button provides additional education about the attack or method to prevent it. The *Hint* button provides useful hints for the player, helping to highlight the area of code which is pertinent to solving the level. The *Return to Main Menu* button returns the user to the game's main screen where levels can be selected.

The left sidebar contains two boxes, *Submission* and *Response*. The *Submission* box is to be used by the player to place the plaintext that he or she deems as the answer to the module. Upon clicking the *Grade Me* button, the plaintext is sent securely using a wss:// socket connection to the Fedora Scoring server, where the data from the user is analyzed and scored. The *Response* box echoes the results and allows the player to see what the player has inputted for submission.

The Main page for the game presents users with a couple of different options. They can select to start from the beginning or level one. They can select the *Continue from Last Point* button where it will bring them to the last level they had played, but not completed. The level selection buttons can be selected to bring them to the exact level they want to play. The buttons will be disabled and grayed if the player has not played that level yet. They can only select a level's button by playing the previous levels to completion and receiving a score for the previous level. Once a level is selected, a player is presented with a loading screen while the VM's are started for that particular level. Figure 7.3 provides a view of the Main page.

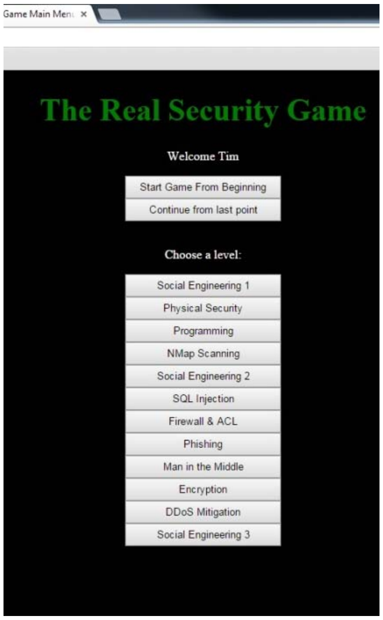


Figure 7.3. Main Page View

7.6. MySQL Server

The MySQL server stores all player information for the game. The player login is stored in the MySQL server with the password hashed as well as a salt for the password. There is also a login_attempts table used along with the PHP login script. This table is used to check for brute force attacks. The table is used to store user_id numbers and the time of the inserted entry. PHP uses this table to check if there have been five failed login attempts in the last two hours. If so, the player is locked out until two hours have passed or the administrator for the game has removed these entries from the table.

Each player is given a unique ID number in the database that is automatically incremented each time a user is added to the database. This ID is used as a Primary key in the database. Each level has its own table as well. The tables for the levels have the foreign key as the player's ID number. That way, it can be referenced when entering scores into the level for the player. The level table also has a *Played* column, which is used to indicate if the player has played the specific level or qualifies to play the level in question. This determination is given to the PHP frontend to grant or deny the player access to the specific level. To help secure the MySQL database, different user-types are created between players and administrators. Each user is given the minimal permissions as the user needs to complete their tasks. Figure 7.4 illustrates the MySQL architecture used.

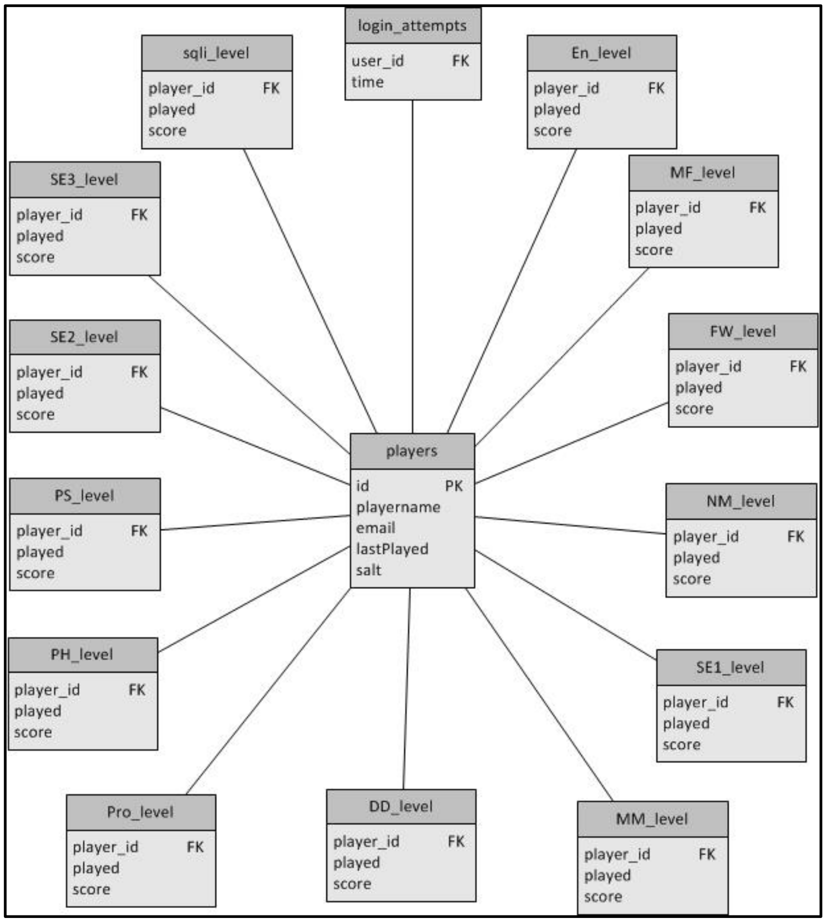
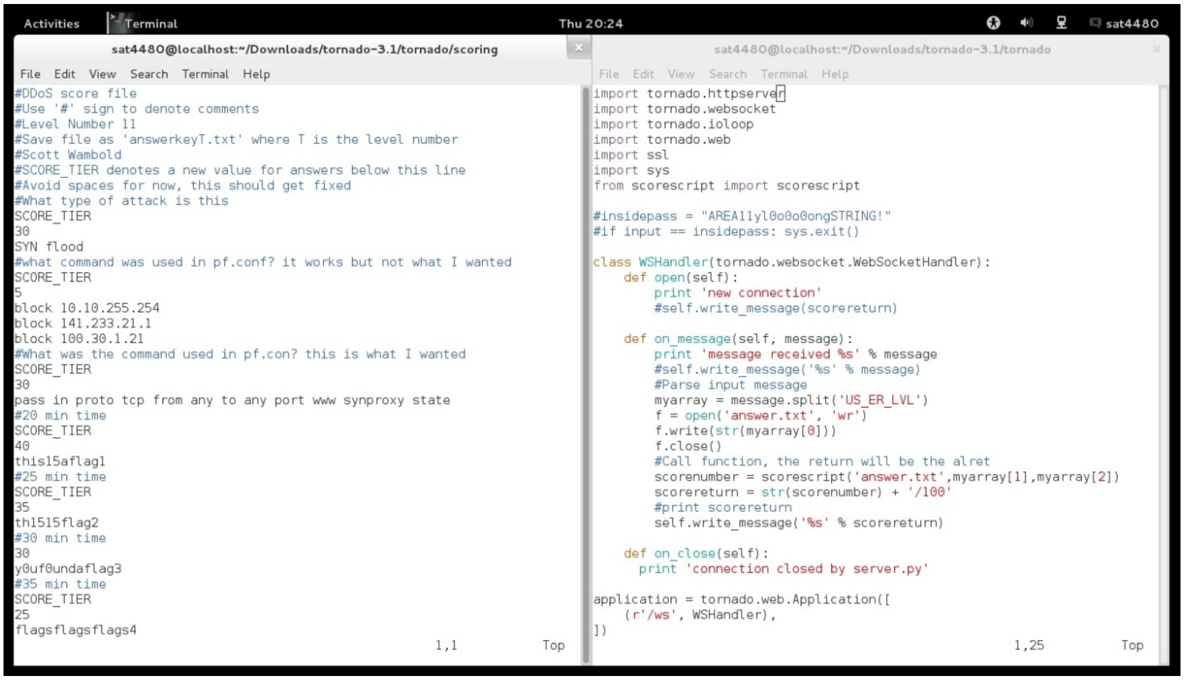


Figure 7.4. MySQL Architecture

7.7. WebSocket Server

The WebSocket server, which was built on top of the scoring server, was built using Tornado version 3.1 as the backend. Early attempts were made to use version 2.0, however it was never successfully implemented. Tornado is a Python Web framework and asynchronous networking library. Once installed, the server code, built in Python, was modified to accept connections on different ports and used a specialized “handler” to be embedded in the JavaScript on the frontend. For this configuration, port 8888 and “/ws” was used. An example of the WebSocket configuration file used to score the defense of a Distributed Denial of Service (DDoS) attack module can be seen in Figure 7.5.

The WebSocket has three main functions: creating the connection, sending a message, and closing the connection. For simplicity purposes, we designed the button *Grade Me* to perform all of these functions. This was implemented by referencing the button id *Connect* for all three functions in our echo JavaScript file.


Activities
Terminal
Thu 20:24
sat4480
sat4480@localhost:~/Downloads/tornado-3.1/tornado/scoring
File Edit View Search Terminal Help
sat4480@localhost:~/Downloads/tornado-3.1/tornado
File Edit View Search Terminal Help

```
#DDoS score file
#Use '#' sign to denote comments
#Level Number 11
#Save file as 'answerkeyT.txt' where T is the level number
#Scott Wambold
#SCORE_TIER denotes a new value for answers below this line
#Avoid spaces for now, this should get fixed
#What type of attack is this
SCORE_TIER
30
SYN flood
#what command was used in pf.conf? it works but not what I wanted
SCORE_TIER
5
block 10.10.255.254
block 141.233.21.1
block 100.30.1.21
#What was the command used in pf.con? this is what I wanted
SCORE_TIER
30
pass in proto tcp from any to any port www synproxy state
#20 min time
SCORE_TIER
40
this15aflag1
#25 min time
SCORE_TIER
35
th1515flag2
#30 min time
30
y0uf0undaflag3
#35 min time
SCORE_TIER
25
flagsflagsflags4
                                                     1,1           Top
```

```
import tornado.httpserver
import tornado.websocket
import tornado.ioloop
import tornado.web
import ssl
import sys
from scorescript import scorescript

#insidepass = "AREA11yl0o0o0ongSTRING!"
#if input == insidepass: sys.exit()

class WSHandler(tornado.websocket.WebSocketHandler):
    def open(self):
        print 'new connection'
        #self.write_message(scorereturn)

    def on_message(self, message):
        print 'message received %s' % message
        #self.write_message('%s' % message)
        #Parse input message
        myarray = message.split('US_ER_LVL')
        f = open('answer.txt', 'wr')
        f.write(str(myarray[0]))
        f.close()
        #Call function, the return will be the alret
        scorenumber = scorescript('answer.txt',myarray[1],myarray[2])
        scorereturn = str(scorenumber) + '/100'
        #print scorereturn
        self.write_message('%s' % scorereturn)

    def on_close(self):
      print 'connection closed by server.py'

application = tornado.web.Application([
    (r'/ws', WSHandler),
])
                                                     1,25          Top
```

Figure 7.5 WebSocket Configuration File

For troubleshooting purposes, additional lines of code were added to the server1.py file, sending a message to the console when a successful connection was made, which text was sent, and when a connection was closed. These lines are currently commented out to avoid them from showing up, but can be uncommented to troubleshoot any problems. For security purposes, an additional line of code was added to immediately close the connection once the message was received. When the server code finishes, and the WebSocket server is started, successful connections from other virtual machines to the WebSocket server are made. It should be noted however, we experienced a "connection refused" when the first attempt was made to send a message from the Web interface to the WebSocket server. Upon troubleshooting, it was determined a firewall rule allowing TCP connections on port 8888 was needed. Once this change was made, the Websocket server was able to successfully receive transmissions from the Web frontend.

Once the WebSocket configuration and functionality was successfully implemented, the next step was to determine how to handle the message received. To tackle this problem, it was necessary to place the received message into a text file on the server for scoring and analyzing purposes, which is described in Scoring section.

8. SCORING

The Scoring server comprises of a Fedora operating system which hosts a MySQL database and WebSocket server. The database tracks user login attempts, as well as module level and user's scoring. The WebSocket server securely sends and receives information from Web clients. The scoring is completed by comparing the results to entries in a MySQL database. If the entry matches, then the corresponding points in each row are given to the individual. Comparisons are also done using the "AND" and "OR" operations to ensure entries are given fair evaluation. Scoring was designed, permitting all levels were able to score using the same method. Players' answers are sent to a scoring server virtual machine using the Tornado Websocket. The answer is stored in a string along with their user id and number. The string is then parsed and the answer is saved while the user id and number are used to start the scoring script. The scoring script will compare the answer file to an answer key configured by the level designer.

The answer key allows designers to choose keywords an answer should contain, as well as each keyword's value, and penalized guessing. The script will then perform error checking, preventing users from scoring outside of the 0 to 100, and store the total values in the database. The score is then output to the user, as well as the option to *try again* or *continue*. In future releases, we plan to add more abilities to the scoring script, including the use of or statements for similar answers, along with testing security and stability.

9. ADDITIONAL SECURITY MEASURES

To protect the game's integrity and accuracy, enforcing security within the game is critical. NoVNC was used as the main connection between the Web clients and the virtual machines and provides one of the most vulnerable areas for attacking the game itself. The NoVNC connection uses SSL encryption to protect against tampering. It also provides a flash-based WebSocket emulator for any browsers that do not fully support WebSockets.

The WebSocket is used to securely provide a socket for communication to and from our module virtual machines, as well as the scoring server. The benefit of WebSockets is that the handshake only needs to be completed once, for the communication to continue. This results in less overhead traffic, as well as less checking of credentials on each connection. WebSockets use the same TLS/SSL security as HTTPS to protect against attacks trying to mimic the module virtual machines, and pass them off as authentic. This encryption also prevents replay attacks from "copying" another user doing the exercises.

Each module virtual machine is secured by enabling only the WebSocket proxy port outside of the firewall. This allows only the secured wss:// connection to be the only way to interact with this virtual machine. If players alter a machine's configuration or render a virtual machine unusable, the machine will be brought back to its original configuration using the VMware workstation's snapshots of VM. The scoring server itself is protected by only allowing wss:// connections that use a certificate signed by the root Certificate Authority (CA).

Logins to the game is conducted using a PHP script that checks the MySQL server for user login information. User passwords are stored using SHA512, along with a hashed salt value. The login script also checks for brute force attacks against a user's account. Just like HTTPS, a certificate was created and installed on the scoring server, using openssl, to encrypt the WebSocket communication. This prevented the player's transmission from being sent in plain text to the scoring server. In order for the WebSocket server to know to encrypt the data, two lines were added that point to the location of the installed certificate.

10. PRELIMINARY RESULTS

The purpose of this project was to develop an interactive and fun computer and network educational game for students targeting a specified age range. Although we are still developing the game, we wanted to gather feedback regarding the approach and design we have taken. To do this, we let fifteen students within the ages of 14-18 years of age play the game. The average results from our questionnaire are shown in Table 10.1. These early results are positive, especially considering the level of students' interest in the field of computer and network security before trying the game and the increased interest after playing the game. There remains room for improvement within the game's user interface, graphics, and response time of the game. We plan to address these areas and build a better experience for the user in the next development cycle.

Table I0.1 Average Response from 15 Students from Player 14-18 years of age

Computer & Network Security Game Survey	Scale from 0-5 (0=No Interest or Poor to 5=Very Good or Great)
Your interest in the computer & network security field before the game.	2.7
Your interest in the computer & network security field after the game.	4.3
Your level of interest in interactive games.	3.9
I would continue to use this game.	4.0
Quality of the computer and network security content.	3.4
Quality of the user interface and graphics.	3.1
I was satisfied with the response time of the game.	3.8
The overall challenge of the game's levels was adequate.	4.2
I found the education and hint sections helpful.	4.1
I learned from each level played.	4.2

A combination of ten teachers who instruct computing courses for students within the targeted age range were also asked to play the game and provide evaluation and feedback. The average responses are summarized in Table 10.2. Results illustrate teachers of the targeted age range would use the computer and network security game as a learning tool within their computing and networking courses.

Average scores for both student and faculty/teachers were positive, however, improvement in areas such as the game's graphical content and overall response time is needed. Open-ended responses from teachers described how they enjoyed the safe environment of the game for the students to play and the education sections within each module, however, many had concerns regarding the lack of computer and network security coursework content within their schools. In essence, they feel the game provides a good introduction, awareness, and platform for students to learn about computer and network security, yet most schools do not have textbooks or other coursework to help reinforce these concepts. Another concern was the training or education of the teachers in the continuously evolving subject of computer and network security. Some teachers felt they did not have adequate knowledge within the game's subject areas if the students had follow-up questions. Each of these two concerns are addressed in the next section.

Table I0.2 Average Response from 10 Teachers Instructing Computing Courses for Targeted Age Range

Computer & Network Security Game Survey	Scale from 0-5 (0=No Interest or Poor to 5=Very Good or Great)
Quality of the game content for computer and network security.	4.1
Quality of the User Interface and graphics.	3.3
Your satisfaction with the response time of the game.	3.1
The overall challenge of the game's levels was adequate.	4.1

I feel the education and hint sections were helpful.	4.8
The education section was useful in my teaching of the respective computer and network security subjects.	4.3
I would use this interactive game as a learning tool for my students.	4.3
I enjoyed using this tool.	4.6

11. CONCLUSIONS AND FUTURE WORKS

Our objective of creating a prototype interactive computer and network security game, which attempted to captivate students while providing rich-education through a fun experience, was met with positive and encouraging results. Both targeted students and teachers who instruct computer-related courses and found the game overall beneficial and would use it as a learning tool for computer and network security education. The major improvements needed consist within the graphical user-interface and enhancement of the response-time of the game. We also plan to explore the development of supplemental coursework in this area for teachers to add to their existing curriculums helping to better meet the teacher's needs while producing content to help reinforce the concepts learned within the game.

Moving forward we will continue to make improvements with the game and continue to test both students and teachers. We also plan to extend the storyline and create additional modules containing more computer and network security educational tools, while we look at options of deploying the produce in a cloud system capable of hosting the virtual machines. Overall, we are pleased with the game's progress and its potential to help educate students in this critical field, and we are excited to continue its development to offer a unique security education platform in the area of system and network administration.

REFERENCES

CHAPMAN, P., BURKET, J. and BRUMLEY, D. "PicoCTF: A Game-Based Computer Security Competition for High School Students". 2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14), San Diego, CA. 2014. USENIX Association.

CHECK POINT SOFTWARE TECHNOLOGIES LTD. "The Risk of Social Engineering on Information Security: A Survey of IT Professionals". Dimensional Research. September 2011. www.checkpoint.com/press/downloads/social-engineering-survey.pdf

CISCO SYSTEMS,INC. "Cisco 2014 Annual Security Report", Cisco Systems, Inc. January 2014. https://www.cisco.com/Web/offer/gist_ty2_asset/Cisco_2014_ASR.pdf

CSOMY, L. "Careers in the growing field of information technology services," Beyond the Numbers: Employment & Unemployment, Vol. 2, No. 9 (U.S. Bureau of Labor Statistics, April 2013), http://www.bls.gov/opub/btn/volume-2/careers-in-growing-field-of-information-technology-services.htm

HACKER HIGH SCHOOL - ISECOM. "About Hacker High School: http://www.hackerhighschool.org/about.html, 2000 [Dec. 12, 2013].

JORDAN, C., KNAPP, M. AND MITCHELL, D. "Countermeasures: An Interactive Game for Security Training" http://Web.cs.wpi.edu/~claypool/mqp/counter-measures/final-paper.pdf, March 2011 [November, 2013].

MARTIN, J. "NoVNC" http://kanaka.github.io/noVNC/ [Dec. 12, 2013].

PHP VIRTUAL BOX - imoore76 "phpVirtualBox" http://sourceforge.net/p/phpvirtualbox/wiki/Home/ [Dec. 12, 2013].

RealVNC - RealVNC.com http://www.realvnc.com/, 2002 [Dec. 12, 2013].

SRIKWAN, S. "Using Cartoons to Teach Internet Security" http://markus-jakobsson.com/papers/jakobsson-cryptologia08.pdf, July 2007 [Dec. 12, 2013].

VMware WSX - VMware "VMware Workstation 9.0.1 Release Notes" https://www.vmware.com/support/ws90/doc/workstation-901-release-notes.html, Nov 2012 [Dec. 12, 2013].

VMware vSphere Web client - VMware "Install and Start the vSphere Web Client" http://pubs.vmware.com/vsphere-50/index.jsp?topic=%2Fcom.vmware.vsphere.install.doc_50%2FGUID-74AA3EF1-BDF3-4752-89DB-A522CDE30A66.html [Dec. 12, 2013].

Wang, X., Hembroff, G. C., Bai, Y., 2013 USENIX Summit for Educators in System Administration (SESA '13), "ITSEED: Development of Instructional Laboratories for IT Security Education" USENIX, the Advanced Computing Systems Association, Washington, D.C., USA. (November 5, 2013).