**--slide 1—**

# INTRODUCTION

***< Each presenter introduces themselves >***

Our presentation today will be on the topic of security, specifically the Importance of Teaching Malware.

**--slide 2—**

Throughout the presentation we will go over a brief history of security, then delve into the current efforts in teaching malware. We will discuss some of the people currently teaching courses on malware, the importance of ethics, and explore am example of a course outline from John Aycock.

After a mystery activity we will finish up with a brief summary, discussing some of the current major roadblocks, and the hopes for the future of teaching malware.

As we get started with the history portion we will also have a handout you can look over during the presentation. It has a few fast facts about malware and a few points about the climate surrounding teaching malware.

**--slide 3—**

# HISTORY

Before we go into detail on why it’s important to educate developers by constructing malware, we think it’s important to look at history to see how security has evolved and in many ways how it is the similar. By looking at history we can see what is necessary to do in order to succeed in combating these new forms of security threats.

**--slide 4—**

The core concepts of security have never changed. When new ways of harming others are developed we attempt to counter it with some other form of technology.

The earliest examples of this are spears and clubs. We needed to develop a way to counter these technologies that could harm us and we ended up developing shields. When swords were first made they were incredibly powerful and we needed some way to defend against them, one of the main ways was heavy metal armor that could reduce the cutting power swords had.

**--slide 5—**

A great example of security from our past were castles, and in many ways they can relate to how a computers security works. One of the main reason a castle was so secure is because of its walls, they could keep unwanted threats outside. This is very similar to how a computers firewall works.

Castles also generally had a garrison or guards that protected the contents inside the castle if anything managed to penetrate it by finding and eliminating threats. This is also how an antivirus works on your computer.

**--slide 6—**

Why does all of this have an impact on us today? You can see from the castle example that many forms of security do not change, they just adapt the same methods within a new environment such as our computers. It will continue to advance in this way because there will always be someone with a malicious intent and they will look to the newest technology to achieve their goals.

**--slide 7—**

Throughout history it has been necessary to study whatever harmful thing comes our way in order to better defend against it. We do not do this nearly enough with software developers because there is a lot of controversy over teaching people to design things that are malicious, but there is no better way to understand the inner workings of something than to build it yourself.

If we look at history it’s clear we need to truly understand what we are up against in order to defend against it. In the present there is a dearth of courses that delve into how malware works.

**--slide 8—**

# CURRENT EFFORTS

A few people of education have realized this and have introduced new methods of learning in the fields of computer science. Their main goals are common in nature. We took the liberty of contacting these professors via email to ensure that we could deliver their message clearly to you all today. Their responses were more than helpful. Let’s take a look at them now…

**--slide 9—**

*John Aycock:*Associate professor of the Computer Science Department at the University of Calgary.

*George Ledin:*Professor of the Computer Science Department of Sonoma State University.

*Dr. John Sullins:*Assistant professor at Sonoma State University (partnered with George Ledin).

**--slide 10—**

Professor John Aycock teaches a course developed in 2005 called “Spam and Spyware” that we will highlight later on. At the time of its development it was the only one known in existence. Similar to Ledin’s theory on learning the development process of malicious code, Professor Aycock believes the more that is known about malware, the better prepared we will be to defend against it. Also he stresses that places of higher learning should be more responsible for handing out education on this subject matter as it is very difficult to find specific resources alone.

**--slide 11—**

George Ledin, a professor in the computer science field who is a trailblazer on malware design and defense education. His course is called “Computer Security and Malware“at Sonoma State University in California. He has been subject to the media for some time due to his involvement with the course. Antivirus companies have shunned him calling “him” the nuisance instead of the malware that they aim to prevent. On the other side, Ledin believes that antivirus products are of no worth whatsoever.

**--slide 12—**

Dr. John Sullins, his responsibilities lie in teaching computer ethics to classes alongside Mr. Ledin.  He specializes in philosophical issues of artificial intelligence/robotics, engineering ethics, philosophy of technology, and more.  He is also a certified Military Master at Arms and winner of the 2011 Herbert A. Simon Award.  Professor Ledin and Dr. Sullins agreed before beginning the curriculum that ethics would need to be stressed given the circumstances of the content.

**--slide 13—**

Before we dive into the malware specific courses offered by some of these professors, we will look at the side requirements that are essential to them.  Ethics.  Dangerous knowledge is how Dr.John Sullins describes the act of malware programming and in cases where it’s taught requires special consideration on ethics.  He alone is responsible for embedding concepts of right and wrong into the student minds that attend Professor Ledin’s computer science program.  Learning the art of malware in his words require firewalls of both technical and moral nature.

**--slide 14—**

Why is this necessary you might ask? Consider from the brief history we looked at and what capabilities some of the malware possessed. If you were taught a method on how to bypass every single antivirus program created, would you be tempted to create the software and utilize it for your own purposes? Depending on your own morals the direction is either way. In order for education to be successful with its malware teaching endeavors, ethics must reinforce the minds of students who will be planted deep in its roots.

Working with malware even for the best of intentions still requires someone to think like the developer who created it. Human nature would say the more you think like a person who developed a program to steal information or break security measures, the more your own judgment changes. A skill ethics can develop is to keep those two worlds separate in one’s own mind.

**--slide 15—**

Now some example methods on how ethics are taught to students.

Basic concepts:

Starting place for students is the ACM Code of Ethics. This is a code that contains 24 crucial statements describing ethical issues a professional may come across in his or her IT career. The code uses a set of guidelines that complement the statements directing how one should proceed upon confronting an issue. Think of it as a programmers “Prime Directive “if you’re a star trek fan. There are areas that the code still may not cover completely so other theories are lectured such as utilitarianism, deontology, human rights, and the unified common goods approach as described by James Moor. These in turn also have their own good and bad areas so other systems are used to cover them.

Virtues in Security:

A concept taught to students are the three virtues on secure software. They are confidentiality, integrity, and availability (CIA). A particular CIA approved analogy on firewalls used to build systems is focused on as well, noting that it is questionable. This is due to the possibility of unknown perils from inside the system. The ethics on data level security are then discussed including the challenges that follow.

Ethical Hacks:

It’s advised that students should not just think like a goodie-good but as a researcher striving to attain different methods that will work for humanity in the future. Also that defying limits and abilities of computer systems is not completely wrong. Innovation thrives on new discovery and a student may very well come across such a discovery. The main concern in the motivation of the student so concentration is applied to their virtues and personal motives. Students ultimately decide whether to take paths of good or evil based on these and not because they just do not choose to follow a basic code of conduct.

Assessments:

The reason why students choose a particular method based on their morals is examined. They are tested in two ways. The first is in a classroom through discussion and contemplation and the second method uses an exam process. While working on projects that involve awkward situations, students will be required to provide their own ethical reasoning for their choices. This is extremely important within the field of malware research.

**--slide 16—**

Now that we have ethics covered we can take a look at the guts of one course being offered by Professor Aycock with the University of Calgary. The courses taught by Professor Aycock and Ledin aim for the similar goals of advancing knowledge on malware to better defend against it. In the words of Professor Ledin:

“The goal is for students to use their knowledge of the “dark side” of programming to build future computer systems that are better equipped to guard against and even combat these malicious programs.”

**--slide 17—**

John Aycock Course

Computer Science 528 Spam and Spyware

Spam and other unsolicited bulk electronic communication, and spyware. Legal and ethical issues. Countermeasures and related security problems. Course Hours: H(3-0)

Prerequisite(s):

Computer Science 313 and 457 and consent of the Department.

Now a brief look at the course called “Spam and Spyware” offered by Professor John Aycock at the University of Calgary. There are many parts to this course so we very briefly summarize each part and what it entails.  We then take a look at the assignments that a student would be required to complete.

Before we continue further here is a short video with Professor Aycock speaking about his course

***< Play John Aycock video >***

**--slide 18--**

Here we have the course divided into slices shown on the pie graph. We will go through the slices with you now.

* Introduction (3%) ( Laboratory protocol, Legal Agreement, Professionalism )

Lab Protocol is about behaviour in the lab. Applied to students, teachers and anyone involved in its maintenance. Treatment of the lab is like a biohazard area. Legal Agreements signed by the students. Professionalism due to imagery that is found in the same environments of spam and spyware. Students are warned ahead of time to deal with it appropriately.

* Definitions of spam and spyware (3%)

What constitutes spam and spyware. An overlook at all software to see if it would fall into the category. Official definitions are supplied later.

* Ethics (8%) ( General ethical theories, recognizing ethical problems, ethical decision making, sample ethical problems, professional codes of ethics and conduct, ACM code of Conduct, IEEE, Canadian Marketing Association )

Assumed that students have little to no training in ethics. Start with general ethical theories. Progresses into more specialized codes. Written ethics assignment is completed.

* Spam and spyware law (11%) ( Canada, Australia, United States )

Laws change rapidly. Examine existing and future legislation. Any cases of breaking laws are noted. Ethics are presented prior to any programming. This ensures a more secure environment by bringing attention to students on their possible actions.

* Spyware (23%)

Covers history, anti-virus and anti-spyware vendors, why it exists, how it gets onto a computer, spyware capabilities and countermeasures, keylogging defenses, startup hooks, hiding and forms of obfuscation, and much more

* Phishing ( 15% )

Teaches history, social engineering, specialized forms, URL tricks, pharming methods, infrastructure for phishing, anti-phishing techniques, and much more

* Fraud (4%)

Advance fee fraud (419 scams), various types of online scams and money laundering

* Email (7%)

Mail system architecture, routing, DNS, SMTP transactions and mail envelopes and headers

* Spam (12%)

Primer on spam while using real life examples, amassing email addresses, anti- harvesting techniques and harvester countermeasures, cleaning and verifying email lists, bulk email software techniques, open relays open proxies, zombies and much more

* Anti-Spam (15%)

Manual spam tracking methods, white and grey listing, tarpits, proof-of-work systems, sender policy framework, filter-evasion used by spammers and much more

**--slide 19—**

So now we have gone through the main segments on the course. There are also five assignments that are part of the Spam and Spyware course taught by Professor Aycock. One of which is an ethics assignment that we touched on a little and is completed prior to the four that are programming based. So starting with…

Assignment 1: Ethics Assignment

A written assignment completed by the student on ethics after the ethics portion of the course is taken.

Assignment 2: spyware/offensive.

Writing spyware that installs a startup hook( basically starts when the operating systems turns on ), changes the browser start page, and performs keylogging. Keylogging is directed at the capture of the username and password used in the web browser to access a ﬁctitious bank’s web site.

Assignment 3: spyware/defensive.

Students exchange their spyware from the previous assignment. They then develop anti-spyware software that accurately detects, identiﬁes, and removes all spyware samples.

Assignment 4: spam/offensive.

Writing bulk mailing software that delivers messages directly to an SMTP server, optionally routing through an open proxy. Because laboratory constraints preclude students from sending a message to multiple recipients in any meaningful way, students instead sent multiple messages to one recipient. Spam and ham corpora (a subset of SpamAssassin’s structured set of texts for testing) are supplied for the students to transmit.

Assignment 5: spam/defensive.

Once the email is delivered to some lucky recipient from the prior assignment, students develop a spam ﬁlter that sorts the recipient’s mailbox into spam and ham messages as accurately as possible.

**--slide 20—**

Much detail is placed into the course provided at the University of Calgary. So now that you have an idea of what knowledge is gained, where do you go from here? What does the future hold for someone with these unique abilities to counteract malicious software? How will this ever help humanity?

***< Sucker activity >***

**--slide 21—**

# CONCLUSION

The past contains many analogies that describe the teaching methods of these professors. These include offensive vs defensive technologies in the days of old to our present day battles against biological diseases. There certainly is a common pattern of "study then safeguard" showing in their examples. As they move forward to spread solutions in places of higher learning, it is hopeful that the future of malware is a bleak and dismal one.

**--slide 22—**

In the past the response to threats has been inefficient, patching holes in the defenses after the holes were found, usually by a malicious party. Solutions for malware attacks rarely pre-empt the attack, and for the victims they often come too late. Those that would do good remain in the dust of those that would do evil.

Other fields of study have adopted the practice of examining how threats work through close examination of those threats in safe, supervised environments. Computer science has been reluctant to adopt the same methodology.

While educators like John Aycock, George Ledin, John Sullins, and others are certainly laying the base layer for the road to conquering this shortcoming there are still some major roadblocks to overcome.

**--slide 23—**

First major roadblock is the fear that students will use the knowledge gained in a course about malware to go out and perpetuate malicious software. It is the worry that knowledge of malware will tempt students into unethical actions.

The second roadblock has a significant effect on institutions as well as educators, and that is the fear of complicity. That should our first fear be well founded, the institution or educator will be seen as aiding or abetting the criminal acts of an offending student.

Finally, these first two roadblocks lead to a lack of knowledgeable educators and properly equipped institutions.

**--slide 24—**

Many feel that teaching how malicious software is developed will lead to its students using such knowledge to develop more viruses or worms, or any other malicious code. Similar fears have, in the past, prevented development in medical science, most notably with the ban on dissection of human cadavers for research. Within the computer science field itself that barrier existed around cryptography but has since been overcome and is one of the things commonly taught in classes on security. Often times however these classes are still elective and amount to mostly history classes without any examination of actual source code.

**--slide 25—**

Many institutions, and educators, are worried that teaching students how malicious code is made could be considered aiding and abetting should one of those students go on to use that knowledge to commit crimes. The same concern could be applied to medical research, or in chemistry. In those fields most people would agree educators would not be held responsible for the independent decisions of an individual student to abuse the knowledge they were taught.

It is important we overcome these two fears as have other fields of study. As technology finds its way into more aspects of our lives it becomes increasingly important that we can trust this technology to not be compromised. Once we begin to move past these two fears we can begin addressing the final roadblock.

**--slide 26—**

Many educators currently have little to no experience with malicious software, after all they have gone through the same education system as their students. Sometimes, this lack of familiarity is involuntary, and some educators may wish to teach courses on malware but lack the resources to set up new courses for examining and teaching malware.

**--slide 27—**

It is important to continue and build upon the work being done by the current educators teaching malware-focused courses like the ones we have discussed today. They have set a precedence, provided a running start and we should not squander the momentum.

As courses in malware become more common so too should the topic become less taboo. The worries of ethical violations will lessen as it has in the medical research field as more students complete these courses and go on to not be terrible people.

With the topic become less taboo in the tech world the discussion should become more mainstream, causing the knowledge to diffuse to the general public. We have seen that happen again within medical research in the form of hygiene.

Ultimately it is hoped that studying malware become a standard component in computer science courses in universities and colleges. This will create a better-prepared generation of software developers and programmers applying security approaches that not only patch the holes in our defenses that are already known, but better anticipate future attacks and can adequately defend against them. Improved understanding of how various malware works should result in better personal security practices of the average user, like improved understanding of germ theory resulted in better personal hygiene.

Thank you for listening to our presentation!