Public Keys, One Way Functions and Hard Problems

TryEngineering

Provided by TryEngineering - www.tryengineering.org

Lesson Focus

Security has always been a major focus of computer science research, and with the explosion of Internet use by commerce, the need for secure transactions has taken on more urgency. Most recently, cyber-thieves demonstrated that true security on the Internet is going to require a new level of understanding of how to protect personal data, and more importantly, financial transactions. This lesson introduces two important concepts: public key encryption and one-way functions. It provides an opportunity for students to understand the underpinnings of almost all Internet security: they will come to appreciate that any lock can be eventually broken, and that theoretical computer scientists study 'hard' problems to lengthen the time it will take to break a lock. Note that this is not a lesson in encryption, but in how mathematics is used to secure information.

Age Levels

Intended for ages 11-14.

Objectives

Introduce students to

- The concept of a public key
- How the modulo function is a one-way function
- How the Diffie-Hellman key exchange uses a one-way function
- What computer scientists mean by 'hard' problems

Anticipated Learner Outcomes

Students will be able to

- Practice creating public keys with the classic color model.
- Exchange information with the Diffie-Hellman method using modulo arithmetic.
- Explain why no lock can be completely secured, and that given time, any mathematical 'lock' can be broken.
- Use exponentiation and modulo arithmetic to create cyber-keys.

Alignment to Curriculum Frameworks

See attached curriculum alignment sheet.

Internet Connections

Exploring Careers in Engineering and Technology Video:

- http://ieeetv.ieee.org/educational_activities/solving-real-world-problems-with-computing-exploring-careers-in-engineering-and-technology

Public Key Full Video

- https://www.youtube.com/watch?v=YEBfamv-_do

Modulo calculator

- http://www.miniwebtool.com/modulo-calculator

One way function calculator

- https://www.mtholyoke.edu/courses/quenell/s2003/ma139/js/powermod.html

Recommended Reading

- Public Key encryption: https://en.wikipedia.org/wiki/Public-key_cryptography
- Modulo Functions: https://en.wikipedia.org/wiki/Modulo_operation

- Diffie-Hellman: https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange
- One-Way Functions: https://en.wikipedia.org/wiki/One-way_function

Optional Writing Activity

In this lesson you learned how the one-way modulo function is used to make it almost impossible for a third party to steal a public key. Take this activity in one of two directions: (1) on the Internet, research other one-way functions and propose a candidate that might be better than modulo; (2) explain what computer scientists mean by a 'hard' problem. Give an example other than solving the inverse of modulo.

Public Keys, One-Way Functions, and Hard Problems

For Teachers:
Teacher Resources

Anticipated Learner Outcomes

Students will be able to

- Practice creating public keys with the classic color model.
- Exchange information with the Diffie-Hellman method using modulo arithmetic.
- Explain why no lock can be completely secured, and that given time, any mathematical 'lock' can be broken.
- Use exponentiation and modulo arithmetic to create cyber-keys.

◆ Materials

- Tempera (washable) paint in a variety of colors. (Colored pencils can also be used, but the result is not as impressive.)
- Disposable bowls for brush cleaning (or re-use small metal, glass, or plastic jars).
- Paintbrushes sufficient for each student to have one, max 1 inch brush.
- Newspaper to cover work areas.
- Unlined index cards, sufficient for each team of 3 students to have as many as 10.
- Paper and pencils sufficient for students' scratch work.
- Calculators, one per team of three (optional).
- Small tokens such as coins, stones, tiny toys, or wrapped candy, sufficient to demonstrate modulo arithmetic. If you plan to let the students keep the tokens, make sure you have more than enough for each person to receive one.
- Timers, as simple as a wall clock or as complex as stop watches. Each group of three will need to calculate the time it takes to do a task.

◆ Procedure

Overview

'Public key encryption' is not as hard a concept as it seems to be at first glance. The idea is actually fairly simple: in order for two parties to privately share information, they each need a way to lock up the information. They need to share a 'key' so that only they can decipher an encrypted message. The problem is that as information moves around the Internet, anyone can view it. So how to share a pass-code or key without publishing it on the Internet. An equally essential idea is that any lock can eventually be broken; the focus of current security research is to make it take an extremely long time to decrypt a cipher. Please note that this lesson is not about encryption and decryption: the process of encoding or decoding a message. It is about establishing a secret key, like the password on your email, or the pin number on a bank account. Public key encryption is a technique for making some part of the password public, but making it almost impossible for the secret part of the password to be discovered. Note that the terminology 'public-key encryption' is a bit confusing, because we are not concerned with the actual process of encryption, but only with establishing the key or password.

The scenario to establish with your students is that two people want to allow each other access to their rooms, without allowing anyone else to enter. But they can't speak to each other (e.g. sharing their password) because others are always listening in. By using

a trick of mathematics called a 'one-way function', they can establish a secret password without others knowing how they created it.

The color combination analogy is a powerful way to understand one-way functions. It is best to watch the process in action, so please watch the public key encryption video (https://www.youtube.com/watch?v=YEBfamv-_do). Practice at least once yourself how to mix colors to form secret colors.

It is also important to make sure your students understand how exponents and modulo arithmetic work. Learning these concepts in combination with this lesson's objectives may be a bit much. However, using this lesson to reinforce a math lesson on exponents may prove powerful. Depending on your math objectives, and your students' expertise, you might consider allowing students to use an online calculator, their own calculators, or have them calculate the numbers by hand. Please consider setting up a worksheet appropriate to your classroom needs, and of course practice the arithmetic ahead of time yourself!

This lesson focuses exclusively on the mechanics of public key encryption: the Diffie-Hellman algorithm, and how modulo arithmetic provides a critical one-way function. The over-arching theme of the lesson is that to date, theoretical computer scientists have not found a way to create secret keys that are unbreakable. The best they can do is to have it take such a very long time to discover the key, that it can't be broken in time to be useful.

'Algorithmic Complexity' is a computer science technique for determining how long it will take for an algorithm to produce a solution. 'Hard' problems are those that will take an extremely long time to solve. One-way problems are those that are easy (and often immediate) to solve in one direction. However, the inverse problem, that is, starting with the answer and calculating the starting value, is almost impossible (e.g. takes a dauntingly long time) unless you have some specific information. Often, reversing a one-way problem also involves trial and error, or from a mathematical standpoint, requires trying all possible solutions.

Procedure During First Hour

1. Watch the IEEE Career video with your students. Spend at most about 10 minutes talking about problem solving in general, and how it might apply to encryption. Explain to your students that in order to keep the Internet secure and their own data private, computer scientists are constantly trying to improve on the way that public-key encryption works.
2. Watch the video on public key encryption https:/www.youtube.com/watch?v=YEBfamv-_do up to where the description of color encoding ends (approximately 4:37 minutes in). Have your students use Worksheet 1 to write down the procedure for how to mix colors. You may need to stop the video a few times to get agreement on the procedure.
3. Divide the class into triplets. If your class size is not a multiple of three, this is an opportunity to give an example of modulo arithmetic. Create a group of four or two as necessary. Remember that modulo is simply the *remainder* of a division, which is exactly what you are doing with your class. In the worksheet instructions the group of two or four will need to adjust the instructions a bit. Take care to help them with that.
4. Distribute paint, brushes, jars of water, etc., and have your students try to replicate the paint example using the instructions on Worksheet 1. Note that there is at

least one missing detail. This is not a structured lab, but rather a sketch of a problem-solving activity. To make it work, students will have to realize that the amount of color is as important as the colors they choose. Depending on your time constraints you may point this out at the start or let them discover this themselves. Teams should share information as they go! This is not a competition.

5. Leave time at the end of the hour to discuss their results and watch the remaining portion of the video. Explain to your students that in the second hour they will be calculating modulo arithmetic, and using exponents. You might want to provide a quick review if you have time.

Procedure During Second Hour

1. Review the public key encryption video again. Consider skipping beyond the color demo (e.g. start at minute 4:30).
2. Demonstrate the concept of exponents to your students, using your set of tokens. Use base 2 because doubling is easy to do by hand. Show them that $2^1 = 2$, $2^2 = 2 \times 2 = 4$, $2^3 = 2 \times 2 \times 2 = 8$, heaping tokens as you go. Explain that $3^1 = 3$, $3^2 = 3 \times 3$ etc.
3. Demonstrate the concept of modulo. Use '3' as the base, starting with 9 tokens. Divide the 9 tokens into three groups, resulting in nothing left over. Explain that 9 mod 3 is 0. Have a student write the remainder on a board or flip chart. Now divide 10 into three groups, resulting in one remainder, then 11, and then 12. As you do the division, use the language '10 modulo 3 is…' etc., so it reinforces the concept. Go all the way to sixteen to hammer home the point.
4. Divide the class into groups of three, reviewing the concept of modulo and creating a group of two or four as needed, distribute Worksheet 2, and if you are giving away the tokens do it now.
5. Work with groups individually as they need it. If some groups finish early have them help the others complete the task. Leave time at the end to discuss the conclusions reached by the groups.

◆ Time Needed

2 sessions, at most 1 hour each. In the first hour students design the color experiment, learn about one-way functions, and modulo in particular. In the second hour they practice calculating modulo arithmetic, practice creating public keys, and study how long it takes to break the code, if it can be broken at all within a time constraint.

Public Keys and One Way Functions

Student Resource:

The following information will help you complete the exercises in this lesson.

A cipher is a simple kind of lock. For example, a password is a cipher. It is a secret code that allows two parties (e.g. two people) to agree that they can share information. Only if you know the cipher or 'password' can you be trusted with the information.

Any cipher can be broken. Eventually. This is a fundamental idea in theoretical computer science – that until you can prove that a lock is unbreakable, you cannot claim that it is. The best that computer science can do to protect information on the Internet is to make it extremely time consuming for someone to figure out your cipher. If it takes too long, trying to figure out the cipher becomes useless.

The scenario to imagine is that two people want to allow each other access to their rooms, without allowing anyone else to enter. But they can't speak to each other (e.g. sharing their password) because others are always listening in. If there is a time limitation on giving the secret password, then a third party can't enter the room if they can't figure out the password in time.

The Internet protocols have a format that anyone can access. Otherwise the Internet wouldn't work. This means that if you want to keep information private you have to *encrypt* it with a cipher. But you can't send your cipher on the Internet because it could be intercepted. The paradox led to the idea of public keys.

A public key is a kind of cipher that allows two parties to share information when they cannot establish a private code or password. They need a way to make *public* only part of their shared secret. The key will work if the two individual secrets are combined in a way that produces a one-way function.

A one-way function is a powerful idea in mathematics. One-way problems are those that are easy (and often immediate) to solve in one direction. However, the *inverse problem*, that is, starting with the answer and calculating the starting value is almost impossible (e.g. takes an incredibly long time) unless you have some pertinent information. Often, reversing a one way problem also involves trial and error, or from a mathematical standpoint trying all the combinations of possible solutions. Keeping part of the information needed in a one-way function secret is the basis of public key encryption.

The Diffie-Hellman Algorithm uses the mathematical ideas of *exponents* and *modulo (remainder)* to create one-way functions so that two parties can share some, but not all information. Worksheet #2 shows you how to use exponents and modulo arithmetic.

Algorithmic Complexity is a computer science technique for determining how long it will take for an algorithm to produce a solution. 'Hard' problems are those that will take an extremely long time to solve. Matching paint color and reversing one-way functions are both hard problems because they require trial and error to solve without key information.

Public Keys and One Way Functions

Student Worksheet 1: Mixing Colors

Directions:

In this lesson you will directly experiment with the color example of public key encryption. Using the public key video shown by your teacher, write down the rules for color mixing on the back of this sheet or on a separate piece of paper. Hint: there is an essential component to getting the color right that is not mentioned in the video. What do you think it is? Before you try to create secret colors, experiment with what it means to get the color right. Don't see it, or running out of time? See if others in your class have figured it out!

In your group of three, assign someone to be the 'spy' and two others to be the 'color makers'. When the color makers create their secret color, make sure the others are not watching. Once you've established all the necessary colors, time how long it takes the 'spy' to try to generate the shared public color. Is it even possible?

Questions:

1. What is the missing information in the color mixing technique? Why is this important?

2. Were you successful in creating a public color using the secret colors? In other words, were the color makers able to consistently create it? Was the spy able to create it, and if so, how long did it take?

Public Keys and One Way Functions

Student Worksheet 2: Diffie-Hellman

Practice with Exponents:

In your group, practice calculating exponent arithmetic by hand. Remind each other that exponents are shorthand for multiplying a number by itself, a number of times. For example, $4^3 = 4*4*4$. (Note that computer scientists use '*' for multiplication instead of 'X'.) Make up problems for each other that are *reasonable* to solve, such as 2^5, 4^2, 10^4, 8^3. Make sure everyone knows how to solve these problems.

Practice with Modulo:

In your group, practice calculating modulo. Remind each other that modulo is the remainder when you divide. For example, 11 modulo 4 is 3 because 11/4 results in a quotient of 2 and a remainder of 3. Make up problems for each other that are *reasonable* to solve. If your teacher allows it, use a modulo calculator such as http://www.miniwebtool.com/modulo-calculator to check your answers. You might also want a faster way to find the remainder without doing long division. Here is a hint: long division keeps track of the number of times you subtract the divisor. Some people find repeated subtraction to be easier (and faster) than long division.

Practice Exponents and Modulo:

The Diffie-Hellman encryption method relies on the fact that raising a result to another exponent is equal to multiplying the two exponents. For example:

$$\left(2^3\right)^4 = 2^{12} = 4096 = \left(2^4\right)^3$$

Convince your group that is true, perhaps by multiplying 2 by itself twelve times.

Creating a One-way Function with Secrets:

Try creating a public key using the Diffie-Hellman trick. The public key encryption https://www.youtube.com/watch?v=YEBfamv-_do video uses '17' for the prime modulus and '3' for the generator. These numbers are public. Don't worry about why they were chosen. The first formula below calculates 'P' based on two secret numbers, 's1' and 's2.' Within your team, assign roles of 'code maker' to two people and 'spy' to the third. One code maker secretly picks a number for 's1', calculates 'R1' and shares it with everyone. The second code maker picks a value for 's2', calculates 'R2' and shares it. Your teacher will instruct you on whether you can use calculators or online resources, or whether you have to do arithmetic by hand. Start a timer. Both code makers should be able to calculate 'P' quickly and should note how long it took. Give the spy at most five minutes to calculate 'P'. Was that enough time? Discuss how to speed up the spy's procedure. Discuss how to make it harder for the spy?

$(3^{s1})^{s2}$ MOD 17

3^{s1} MOD 17 = R1

3^{s2} MOD 17 = R2

R1 ^{s2} MOD 17 = P

R2 ^{s1} MOD 17 = P

Public Keys and One Way Functions

Teacher Resource:
Alignment to Curriculum Frameworks

Note: All lesson plans in this series are aligned to the Computer Science Teachers Association K-12 Computer Science Standards, the U.S. Common Core State Standards for Mathematics, and if applicable also to the National Council of Teachers of Mathematics' Principles and Standards for School Mathematics, the International Technology Education Association's Standards for Technological Literacy, and the U.S. National Science Education Standards which were produced by the National Research Council.

◆ Principles and Standards for School Mathematics

Number and Operations

As a result of activities, all students should

- Compute fluently and make reasonable estimates

Algebra Standard

As a result of activities, all students should

- Understand patterns, relations, and functions.
- Use mathematical models to represent and understand quantitative relationships.

Problem Solving Standard

As a result of activities, all students should

- Apply and adapt a variety of appropriate strategies to solve problems.
- Monitor and reflect on the process of mathematical problem solving.

Communication Standards

- Communicate their mathematical thinking coherently and clearly to peers, teachers and others

◆ Common Core State Standards for Mathematics

Expressions and Equations

- Apply and extend previous understandings of arithmetic to algebraic expressions.
  - CCSS.MATH.CONTENT.6.EE.A.1 - Write and evaluate numerical expressions involving whole-number exponents.
- Work with radicals and integer exponents.
  - CCSS.MATH.CONTENT.8.EE.A.1- Know and apply the properties of integer exponents to generate equivalent numerical expressions. For example, $3^2 \times 3^{-5} = 3^{-3} = 1/33 = 1/27$.

Functions

- Use functions to model relationships between quantities.
  - CCSS.MATH.CONTENT.8.F.B.4 - Construct a function to model a linear relationship between two quantities. Determine the rate of change and initial value of the function from a description of a relationship or from two (x, y) values, including reading these from a table or from a graph. Interpret the rate of change and initial value of a linear function in terms of the situation it models, and in terms of its graph or a table of values.

Public Keys and One Way Functions

Teacher Resource: Alignment to Curriculum Frameworks

◆ Standards for Technological Literacy – All Ages

The Nature of Technology

- Standard 1: Students will develop an understanding of the characteristics and scope of technology.
- Standard 2: Students will develop an understanding of the core concepts of technology.

◆CSTA K-12 Computer Science Standards Grades 6-9 (ages 11-14)

5. 2 Level 2: Computer Science and Community (L2)

- Computational Thinking (CT)
  3. Define an algorithm as a sequence of instructions that can be processed by a computer.
- Collaboration (CL)
  3. Collaborate with peers, experts, and others using collaborative practices such as pair programming, working in project teams, and participating in group active learning activities.
- Computing Practice & Programming (CPP)
  4. Demonstrate an understanding of algorithms and their practical application.
  9. Explain the principles of security by examining encryption, cryptography, and authentication techniques.

CSTA K-12 Computer Science Standards Grades 9-12 (ages 14-18)

5.3.A Computer Science in the Modern World (MWJ)

- Computational Thinking (CT)
  8. Use modeling and simulation to represent and understand natural phenomena.
- Community, Global, and Ethical Impacts (CI)
  10. Describe security and privacy issues that relate to computer networks.

5.3.B Computer Science Concepts and Practices

- Computational Thinking (CT)
  9. Analyze data and identify patterns through modeling and simulation.