Cryptography in the Classroom using Maple

A. Baliga and S. Boztas
Department of Mathematics
RMIT University
asha@rmit.edu.au, serdar@rmit.edu.au

Abstract

Increasing use of the internet has resulted in cryptography becoming a much sought after elective for Communications, Electronics and Computer Systems Engineering students in the fourth year of their degree. A very mathematical subject which depends on the high complexity of certain operations, the need is for Engineering students to see the use of cryptography without being bogged down by the mathematics. With the help of Maple we are trying to make cryptography more accessible to the students.

1 Introduction

The teaching of mathematics into other programs, particularly engineering and computer science is a vital source of income for many mathematics departments. In our department about 80% of our funding comes from teaching service mathematics. With the changing environment it is very important to bring in new subjects which address the needs of industry. One such subject which has seen a growth in interest is cryptography.

Industries as diverse as banks, transport networks, hospitals and police now keep records almost solely on computers and networks. The recent spate of media articles on criminal activities like the Melissa or the Love-bug virus, or the case of Microsoft having its secret code files viewed, all attest to the necessity of securing these databases. Furthermore every time a credit card is used on the Internet or in a shop, we presume the security of personal details and the prevention of unauthorized use. With computers getting faster everyday it is becoming more obvious that physical security is not enough. The information itself needs to be secure. This is where Information Security comes in. Thus Information Security is an area which impacts on all areas of life due to the increasing use of the internet. Cryptography is just one facet of Information Security.

A very mathematical subject which depends on the high complexity of certain operations, cryptography could become an esoteric mathematical subject, but that would not serve the needs of the engineering students. Software is used in a variety of innovative ways in the teaching of both our specialist mathematics subjects and our service mathematics subjects, see [3, 7]. These innovations were seen to be a natural way to enhance teaching specialized discrete mathematics to higher year engineering students.

2 Relevance of Cryptography

Cryptography has a mixed history. In the past it has been used primarily by various defence organisations for the purpose of securing communications between allies and spying on enemy communicaions. The most famous example of this is the Enigma machine used by Germany and its allies, primarily during World War II, see [17, 12].

However as communication and computing technologies have matured, new opportunities for electronic business have proliferated. The network infrastructure for these new activities is quite heterogeneous, ranging from public (such as the internet, the Public Switched Telephone Network (PSTN), other public Packet networks) to private (the EFTPOS network, company based but countrywide or even worldwide intranets) to a combination of a number of networks. The same heterogeneity is inherent in the network protocols, which are used in these various settings. The myriad opportunities presented by this flexible network infrastructure have to be balanced against the very real risks to communication such as eavesdropping, tampering with messages and impersonation by malicious third parties. Thus coding and cryptography like many topics traditionally in the realm of pure mathematics have become applied and central to this new area of Information Technology called Information Security.

3 Cryptography at RMIT

Discrete mathematics has been taught at RMIT (mainly to Computer Science and Communication and Electronics Engineering students) since 1987. The teaching commenced with Discrete Maths I, II and III in 1987. MA021, Discrete Mathematics I, and MA022, Discrete Mathematics II, were designed as a sequence, drawing on standard texts in discrete mathematics for Computer Scientists such as [18, 5]. MA023, Discrete Mathematics III, consisted primarily of logic programming and consequently required more effort to develop the lecture materials from several sources.

These introductory courses continuously evolved and they are currently offered as MA915 and MA916, Discrete Mathematics IA and IB and teach relations, Boolean algebra and introductions to set theory drawing from both current textbooks such as [11], as well as our own Lecture Notes in Mathematics Series [6]. We have also progressed to polynomial transforms, logic and networks [1] which are taught to higher years.

The group currently teaching the discrete mathematics is mainly comprised of those involved in research in Algebra. Over the years the research interests of this group has evolved to include coding theory and more recently cryptography.

Realizing the greater need for these skills in the community with the explosion of the internet and e-commerce, the group became a special interest group in Information Theory and Security (ITS) in the Department of Mathematics. In 1999, after consultation with the banking and other Information Technology (IT) industries, the ITS group proposed the post-graduate programs in Information Security which are now in their second year of operation. These post-graduate programs are mainly targeted towards either IT professionals or those who would like to move into this field.

Around this time we also decided to offer cryptography as an additional subject to the fourth year engineering students. MA036, Coding and Cryptography, was on the list of electives since 1997, but in 1999 it was felt that there was a need for a separate cryptography subject, MA038,

Applied Cryptography. Thus teaching of cryptography to engineering and computer science students is relatively new to RMIT.

The design of the postgraduate courses is more industry oriented with the internet forming a major component in the development of course-material. In these courses students are provided with copies of the slides used in the lectures and references, instead of the traditional lecture notes.

The undergraduate subject aims at a more concise syllabus with the aim being to give an overview of the subject with glimpses of the current areas of interest.

4 Use of Maple

Mathematics has traditionally been a pen-and-paper area. Many mathematicians are justly proud of their ability to do mathematics without the use of any computer equipment.

Cryptography involves the study of securing information. Information is encrypted using an encrypting algorithm which is usually public knowledge. Just as a different locks of the same make have different keys, the security of the encrypting algorithm is powered by the encryption key.

One of the main requirements of a cryptosystem is that it should not be vulnerable to anything but a brute force attack. The key length is a measure of how long a brute force attack will take. With the continuous advancement of computing facilities and computing power the current cryptosystems use keys of length varying from 512 bits for very short term use such as smart cards to 2048 bits for certification authorities.

The actual algorithms use various principles of number theory. Many of these principles rely on the difficulty of performing a certain operation such as factoring a large number. These algorithms have not yet been proved to be impossible to crack, just very time-consuming. See for example the Handbook of Applied Cryptography [14] or Coutinho [8]) for more details on integer factorization and algorithms based on it. Of course small numbers can be used to explain the details behind the theory, but the power of the algorithm can only be demonstrated using large numbers with around 20 to 30 decimal digits. With Maple it is possible to work on non-trivial examples. Even an inexperienced student can easily change the parameters in the examples and try out new ones. From past experience of the use of Maple (and Mathematica) in several undergraduate subjects at RMIT [10, 4] it is clear that most students truly enjoy using a computer algebra package. Rarely does the use actually require the students to develop a mastery of the package [13].

5 MA038 - Applied Cryptography

MA038 consists of the study of the design and analysis of cryptographic techniques. The idea is to teach students both a brief history of Cryptography as well as to showcase the current advances in the field. Maple complements the lectures with a tutorial every three weeks.

The first tutorial is an introductory one where the relevant functions are explained and the students are given time to familiarize themselves with the basic notation of the package. The remaining tutorials consist of a live demonstration of pre-prepared Maple worksheet. The students are then encouraged to change various parameters in a supervised session. There

is also a homework component to each tutorial. These assignments (homework components) account for 45% of the assessment in the subject. From student feedback it is obvious that students find these Maple sessions very useful.

One of the first topics handled in MA038 is classical cryptography. Cryptography, the science of securing communication from unauthorized reading, is over 2000 years old. Cryptanalysis, the method of breaking into cryptosystems is considered by some to be the second oldest profession in the world. Until around 1945, cryptology, the science of secrecy, was used almost solely for military and/or diplomatic purposes.

One of the oldest cipher used is credited to Julius Caesar and hence is called the Caesar cipher or Caesar shift. It shifts letters in the text in a cyclic manner over k places. For example if $k = 5$, the word cipher would be encrypted as follows:

cipher $\xrightarrow{+1}$ djqifs $\xrightarrow{+1}$ ekrjgt $\xrightarrow{+1}$ flskhu $\xrightarrow{+1}$ gmtliv $\xrightarrow{+1}$ hnumjw

The Caesar cipher is an example of a monoalphabetic substitution cipher, meaning that all the letters are shifted by the same k. Thus only k, the key, is needed to break the cipher.

5.1 Substitution Ciphers

The simplest kind of substitution is the monoalphabetic substitution, such as the Caesar shift or any rotational shift, see [19]. But simple rotation is insecure — only the shift number, the key, needs to be found to crack the encryption. The next technique is to use a key word to start the permutation. But the word pattern remains, and hence can be attacked by substituting small words first and using knowledge of the language.

To overcome the vulnerability of monoalphabetic ciphers we can use polyalphabetic ciphers, see [19]. Here is an example of the Maple tutorial we use to help the students understand the substitution cipher and the functions required to change algorithms into computer-recognizable code.

First we need a function to convert the letters and symbols of the English alphabet into ASCII code. The letters and numbers are represented by the decimal numbers from 32 to 128 in ASCII. Let L be the list of the letters and symbols. The function "convert" converts these into ASCII. Note that the parameter "bytes" is bidirectional converting from ASCII to symbol and back.

$[> \mathbf{L} := \mathbf{convert}([\mathbf{seq}(\mathbf{x}, \mathbf{x} = \mathbf{32}..\mathbf{128})], \mathbf{bytes});$

The list of ASCII numbers from 32 to 128 are converted into symbols.

$[> \mathbf{convert}(\mathbf{L}, \mathbf{bytes});$

reconverts the symbols into ASCII.

An example of the polyalphabetic substitution cipher is the Vignere cipher, see [19]. We can use this cipher to encrypt some plain text. The next tutorial deals with the use of the Caesar shift to cryptanalyze the cipher text.

Let the plain text be 'hereistheplaintext'

$[> \mathbf{plaintext} := \mathbf{convert}(\text{`}\mathbf{hereistheplaintext}', \mathbf{bytes});$

and let the key be the word 'key' repeated as many times as necessary.

[> **keytext** := **convert**('**keykeykeykeykeykey**′, **bytes**);

It is too difficult to remember the positions (or ASCII codes) of the letters in this form. It would be more convenient if the letters were numbered 1 for 'a' to 26 for 'z'. So we work out the number assigned to 'a' which is 97 and subtract 97 from each number.

[> **constantvec** := (**a**, **len**)− > [**seq**(**a**, **i** = **1**..**len**)];

Here *"constantvec"* converts the symbol or letter into a vector of length *"len"*.

[> **convert**(**constantvec**(**97**, **20**), **bytes**);

That is, let the range of letters be $a = 0, b = 1, ..., z = 25$.

The plaintext and key are now changed into this range.
[> **plaintext** := **convert**('**hereistheplaintext**′, **bytes**) − **constantvec**(**97**, **18**);
[> **keytext** := **convert**('**keykeykeykeykeykey**′, **bytes**) − **constantvec**(**97**, **18**);

Now the plain text and the key text can be added to obtain the cipher text.
[> **ciphertext** := **plaintext** + **keytextmod26**;

But if we tried to convert this into English we would only get garbage as these are not the ASCII values of the letter of the alphabet.
[> **convert**(**ciphertext**, **bytes**);

Thus a re-conversion into the appropriate values is required.
[> **ciphertext** := **plaintext** + **keytextmod26** + **constantvec**(**97**, **18**);
[> **convert**(**ciphertext**, **bytes**);
Thus we obtain the encrypted version of the plaintext.

5.2 Modern cryptosystems

As Schneier [16] says, the main idea behind using a cryptosystem is that a group of people could use some private information to keep written messages unreadable to everyone else. Suppose Alice has a message (sometimes called the plaintext) that she wants to send to Bob. She may want to be able to read the message again at some later date. But she does not want anyone else to read this message. So Alice encrypts the message. She invents some way of changing the plaintext message into the ciphertext message. If an eavesdropper (let us call her Eve) gets her hands on this ciphertext, she cannot make out what the message is. This idea works, more or less.

But there are complications. Firstly, the algorithm that Alice is using has to be good. Eve is not going to give up easily, especially if she works for the National Security Agency (NSA), for example. Also you need to be able to add and remove people you want to share your secret with.

In the present age, you may be an internet user and want to be able to communicate securely with many different people. But you do not want to share the same secret — i.e., you want separate secret algorithms. Just like the locks we use on our front doors we want algorithms

which are standard but have different keys. The Data Encryption Standard (DES) has been the standard cryptographic algorithm since 1977. The DES algorithm is well known, but that does not affect its security since each group of users choose their own secret key. Alice and Bob share a key so that they may communicate securely without Eve reading their communication. They can include a new person, say Chris into their group by giving him the key. If, later, they want to exclude Chris, they just change the key without telling him. The DES is an example of a symmetric algorithm, so called because the sender and the receiver share the same key.

Other common symmetric algorithms are the triple-DES, RC4 and RC5, Blowfish and AES, the Advanced Encryption Standard. These algorithms are used to secure many communications, including online banking and e-mail. But they are not perfect. The main problem is in the key distribution.

Public-key cryptography (or asymmetric encryption) solves the problem of Alice and Bob being sure that they share the same key. It also enables you to send secret messages to people you have not met. In 1976 Diffie and Hellman [9] explained the basic idea involved, which is to use a mathematical function that is easy to compute in one direction but hard to compute in the other, i.e. hard to invert without additional information. Thus instead of a single key, Alice and Bob share two keys — one for encryption (computing the function) and another for decryption (computing the inverse of the function). The two keys are different and one cannot be obtained from the other. Bob can create these two keys and he publishes the encryption key. Alice can find this key, use it to encrypt her message (i.e., compute the value of her message using the function) and send it to Bob. Bob then uses his decryption key, which he has kept secret, to decrypt her message. Thus Bob uses the extra information (i.e., the decryption key) to invert the function and hence find the value of Alice's message. The important thing to note here is that Alice does not need to meet Bob, she may not even know him.

In 1978 Rivest, Shamir and Adleman [15] proposed the RSA public-key cryptosystem. It uses the fact that exponentiation modulo a large composite number n is relatively easy to compute but taking the roots modulo n is in general believed to be intractable. See Singh [17] for an easy explanation of the technique.

6 Conclusion

Maple allows students exploration of non-trivial examples of encryption. They also enjoy seeing the actual algorithms used in commercial cryptosystems such as RSA and DES. Maple gives them the opportunity of doing more involved examples and understanding both the reliability and unreliability of different cipher systems. It can demonstrate very well that a large key space should not fool one into believing that a system is secure.

These Maple tutorials and assignments have also been introduced into every course of the Post-graduate Programs in Information Security and are proving a useful tool. We are now introducing MAGMA (a more sophisticated, specialist algebra package) to the more advanced subjects of the post-graduate programs.

References

[1] A. Baliga, S. Boztas and G. T. Clarke, Networks and Polynomial Transforms, RMIT Lecture Notes in Mathematics, Ed. G. F. Fitz-Gerald, RMIT, Melbourne, 1999.

[2] B. Blyth, "Animations using Maple in First year", Questiones Mathematicae, Suppl. 1, Proceedings of the Warthog DELTA '01 Conference, NISC Pty. Ltd., 2001, pp. 201-208.

[3] B. Blyth, "Mathematics for Surveyors using Mathematica in a laboratory", *Waves of Change*, Proceedings of the 10^{th} Australasian Conference on Education Engineering, 5^{th} Australasian Women in Engineering Forum, 5^{th} National Conference on Teaching Engineering Design, eds. P. Howard, G. Swarbrick, A. Churches, Central Queensland University, Rockhampton, Australia, (1998), pp. 319-322.

[4] B. Blyth and J. Shepherd, "Nonlinear mathematics using Maple in first year", Proceedings of the 1998 International Conference on the Teaching of Mathematics, Samos, Greece, John Wiley, 1998, pp. 50-52.

[5] J. Bradley, Introduction to Discrete Mathematics, Addison-Wesley Pub. Co. Inc., 1988.

[6] G. T. Clarke, Boolean Algebra, RMIT Lecture Notes in Mathematics, Ed. G. F. Fitz-Gerald, RMIT, Melbourne, 1998.

[7] H. Connell, B. Blyth, R. May and C. Zorzan, "Teaching the Finite Element Method using software", *The Challenge of Diversity*, proceedings of the Delta'99 symposium on Undergraduate Mathematics, ed W. Spunde, P. Cretchley & R. Hubbard, Delta'99 Committee, Central Queensland University, Rockhampton, Australia, 1999, pp. 65-68.

[8] S.C. Coutinho, The Mathematics of Ciphers, A.K. Peters, Ltd., Natick, Mass., 1999.

[9] W. Diffie and M.E. Hellman, "New directions in cryptography", IEEE Trans. Info. Th. IT-22, Nov 1976, pp. 644-654

[10] G. Fitz-Gerald and L. Healy, "Serving Maple using WWW", Proceedings of the 1996 International Conference on Multimedia Engineering Education, ed. M. Aldeen, The University of Melbourne, Australia, 3-5 July, 1996, pp. 409-418.

[11] R. Johnsonbaugh, Discrete Mathematics, 5th Ed., Prentice Hall International, Upper Saddle River N.J., 2001.

[12] D. Kahn, The Code Breakers, Weidenfeld and Nicolson, London, 1974.

[13] M. May, "Designing courseware with Maple, (without expecting the students to learn Maple)", Proceedings of the Twelfth International Conference in Technology in Collegiate Mathematics, ed G. Goodell, Addison-Wesley, 2001, pp. 240-243.

[14] A.J.Menezes, P.C. van Oorschot and S.A. Vanstone, The Handbook of Applied Cryptography, CRC Press, Oct 1996.

[15] R.L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Comm. ACM, Vol. 21, Feb 1978, pp. 120-126.

[16] B. Schneier, Secrets and Lies - Digital Security in a Networked World, John Wiley & Sons, Inc., New York, 2000.

[17] S. Singh, The Code Book, Fourth Estate Ltd., London 1999.

[18] R. Skvarcius and W. Robinson, Discrete mathematics with Computer Science applications, Benjamin/Cummings Pub. Co., Menlo Park, Calif., 1986.

[19] H.C.A. van Tilborg, Fundamentals of Cryptology, Kluwer Academic Publishers, 2000.