# A Cryptography Course for Non-Mathematicians

Rich Schlesinger
Kennesaw State University
1000 Chastain Rd. MS 1101
Kennesaw GA 30144
+1 770-423-6039
rschlesi@kennesaw.edu

## ABSTRACT
Traditionally, courses in cryptography have been heavily mathematical in nature. Yet, there is a large population of Information Systems practitioners who are not mathematicians, but who need to implement cryptography as a part of an overall system that they are developing. These people need a thorough understanding of the characteristics of good cryptographic communication protocols. Without this level of understanding, numerous cryptosystems have been deployed that use proper encryption algorithms (e.g. DES, RC4), yet had significant flaws. This paper describes an attempt to structure a cryptography course for IS practitioners.

## Categories and Subject Descriptors
C.2.0 [**Computer Communications Networks**]: General – *Security and protection*
E.3. [**Data Encryption**] - *Data encryption standard (DES), Public key cryptosystems, Standards (e.g., DES, PGP, RSA)*
K.3.2 [**Computers And Education**] - Computer and Information Science Education – *Curriculum, Information systems education.*
K.6.5 [**Management Of Computing And Information Systems]** - Security and Protection – *Authentication, Invasive software, Unauthorized access.*

## General Terms
Algorithms, Security.

## Keywords
Cryptography

## 1. INTRODUCTION
Historically, cryptography has been taught as a highly mathematical subject, utilizing Finite Fields and Number Theory. Certainly, this is necessary for a complete understanding of the underlying mechanisms in modern encryption algorithms. However, the vast majority of people developing or evaluating a cryptosystem will never create or crypto-analyze the underlying encryption algorithms. They are much more concerned with the upper level protocols of the cryptosystem (Key Management and Usage).

Because of this emphasis on the mathematics of cryptography, many students who would otherwise take a cryptography course and who would need knowledge of cryptography as Information Systems practitioners fail to learn this material. This in turn has resulted in numerous flawed cryptosystems in which there was nothing wrong with the underlying encryption algorithm


*InfoSecCD Conference'04*, October 8, 2004, Kennesaw, GA, USA.


## 2. EXAMPLES OF FLAWED SYSTEMS
In this section, we describe several flawed cryptosystems that have been deployed. As we shall see, these flaws are primarily due to a lack of understanding of the necessary communication protocols, rather than the mathematical basis for the underlying encryption algorithm.

### 2.1 Merchant Debit Card Systems
Merchant Debit card systems were deployed in the early 1990s to allow merchants to accept debit cards at checkout. These systems use a "pinpad" to encrypt a customer's pin number (See Figure 1). As originally installed, these systems used the "Master-Session" key management system, which required that the credit card authorization host should periodically download a new session key to be used by the pinpad [1]. Unfortunately, the people who designed the communication protocols used between the credit card authorization host and the merchant terminal were typically unaware of this requirement and did not include a provision in the communication protocols to provide a new session key. Thus, in most situations, the session key was never changed

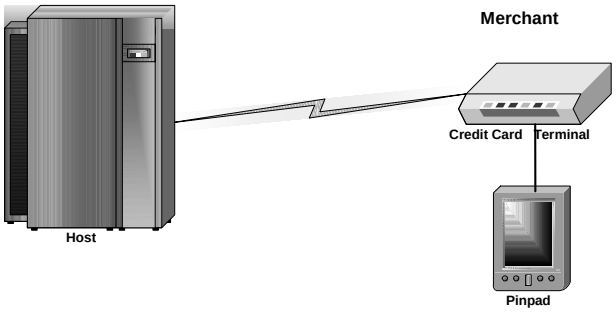



**Figure 1 Merchant Debit Card Authorization System**

## 2.2 IBM 4758 Cryptographic Processor

The IBM 4758 Cryptographic Processor is used in ATMs. Unfortunately, a flaw in the design of the programming libraries for this system allowed a knowledgeable bank employee to obtain the keys that were being used by the device [3].

## 2.3 Wired Equivalent Privacy

Wired Equivalent Privacy (WEP) is a feature that was included with the IEEE 802.11b standard for wireless communications (See Figure 2 Wireless LAN). This standard uses the RC4 cipher for encryption. To prevent RC4 based systems from being compromised; the keys must never be re-used. Unfortunately, the IEEE 802.11b standard never mentions this and many manufacturers developed systems that effectively repeat the key, thus leading to fairly easy compromise of the system [4].

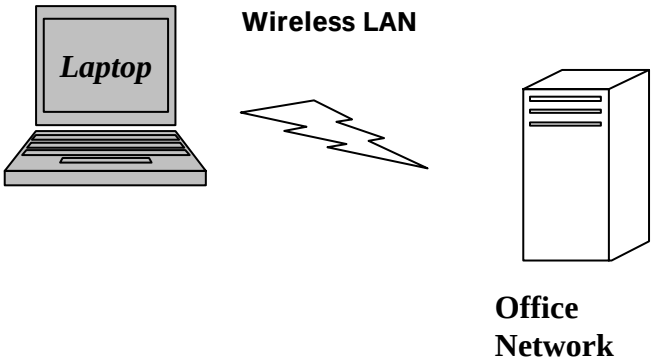



**Figure 2 Wireless LAN**

## 2.4 Microsoft Passport

The Microsoft Passport system is intended as the portal to all of Microsoft's new .NET services. The initial implementation of Passport provided a convenient method for securely storing a user's private information (address, credit card number, etc). That information could then be used when shopping with a Passport enabled web merchant.

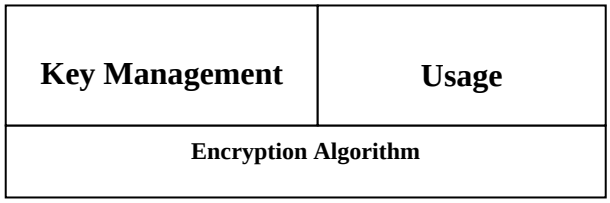
The protocols that Passport uses are a variation on Kerberos. Unfortunately, Passport does not require that each user have a private key. This change in the protocol allows the system to be open to a combination Trojan horse and cryptographic replay attack. Details about this system may be found in [10].

# 3. WHAT IS A CRYPTOSYSTEM?

If one were to ask students what is a cryptosystem, most would say it is the encryption algorithm (e.g. DES or RSA). But it is important to understand that there are two additional components to a cryptosystem.

Key Management – How Keys are chosen and communicated to various systems.

Encryption Usage – How the cryptosystem is actually used to protect data,




If one examines the root causes of the flaws cited above, one quickly discovers that they all used acceptable encryption algorithms (DES, Triple DES, RC4, etc). The flaws were in the Key Management and Encryption Usage. These types of problems require an understanding of proper communication protocols. A deep understanding of finite fields or number theory is not required. It is upon this basis that we set out to structure an undergraduate course with the objective that students would focus on the protocols required for understanding these upper layers of a cryptosystem.

# 4. COURSE MATERIAL

The course intermixes a presentation of cryptographic fundamentals with exploration of real-world systems, including using flawed systems as case studies. It also covers tools that can be used to develop cryptographic systems.

## 4.1 Concentration on Cryptographic Protocols

As mentioned earlier, the course concentrates on discussing the following types of issues that students would likely encounter in industry:

1) Understanding the requirements and implications of the encryption algorithm being used. Thus, the discussion on WEP emphasizes that RC4 requires non-repeating random keys. Discussion on DES covers when to use the different modes of operation.
2) Discussion about how and when to change encryption keys.
3) Examining the overall protocol for Man-in-the-middle attacks. Cryptographic protocols need to include measures such as Message Authentication Codes (a cryptographic hash) to prevent or detect this type of attack.
4) Examining the overall protocol for possible replay attacks. Cryptographic protocols need to be time-stamped and/or tied to communication sessions to avoid replay attacks.
5) Understand information leakage attacks. Message protocols and library APIs should be designed so they cannot be used for unintended purposes. For example, a diagnostic request should only report on the health of the system, not provide details of the system's cryptographic state information (e.g. Key Values).

# 5. STUDENT ASSIGNMENTS

The students in a course such as this will vary from those who have a high interest in programming to those who are more interested in project management. Thus, it was decided to give the students a mix of assignments; some programming projects and

some research papers. The programming assignments were done in Java since students had already learned Java and cryptographic libraries are included in the standard distribution.

The assignments have points (level of difficulty) assigned to them. Each student could choose the assignments they did as long as they did one programming assignment and one paper.

| Assignment | Type | Points |
|---|---|---|
| Quantum Cryptography | Paper | 1 |
| Modes of Operation | Programming | 2 |
| WEP | Paper | 2 |
| Performance Comparison | Programming | 3 |
| Key Recovery | Paper | 1 |
| SSL | Programming | 3 |
| HW Crypto Devices | Paper | 1 |
| PGP | Paper | 1 |

## 6. TEXTBOOK

The choice of a textbook for such a course is critical. The textbook needs to cover the upper level protocols without completely ignoring the underlying encryption algorithms. Two commonly used textbooks are Schneier [9] and Menezes [2]. They were considered too mathematical for this course. The final choice was Stallings [11], which has a good discussion on the underlying mathematics, but emphasizes the network communication protocols.

## 7. COURSE EXPERIENCE

This course was taught for the first time in Spring 2004 as a senior level Special Topics course. The programming assignments were more difficult than originally expected. Most students were using the simple development environment provided to them for their first programming course. This development environment made it extremely difficult to use the additional libraries required for some of the encryption algorithms.

The students completed an anonymous survey questionnaire at the end of the semester. The survey results showed that students liked the "pick and choose" approach to the assignments. In general, the students felt the assignments provided a lot of additional understanding of the material. There were mixed remarks about the textbook, with most students saying that it was understandable, while some felt that it was too difficult. We were quite pleased that the students unanimously said they would recommend this course to others.

## 8. CONCLUSIONS

This course successfully met its objective of teaching students about the necessary communication protocols required in cryptography without getting mired in the underlying mathematics. The combination of the course material and the homework assignments engaged the students so that they now have a good understanding of this material. A future enhancement for this course will be to include laboratory exercise using smartcards.

## 9. REFERENCES


[1] American National Standards Institute. *Retail Financial Services, Symmetric Key Management, Part 1: Using Symmetric Techniques* (ANSI X9.24) Accredited Standards Committee X9, Annapolis, MD, 2002.

[2] Menezes, A., van Oorschot, P., Vanstone, S. *Handbook of Cryptography*, CRC Press, New York, 2001.

[3] Bond, M. & Clayton, R., *Extracting a 3DES Key from an IBM 4758*, http://www.cl.cam.ac.uk/~rnc1/descrack/index.html, 2001

[4] Borisov, N., Goldberg I., Wagner, I., *Security of the WEP Algorithm*, http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html, 2001

[5] Hewlett-Packard, *HP Atalla Security Products*, http://h20138.www2.hp.com/, 2003

[6] Massachusetts Institute of Technology, *Kerberos: The Network Authentication Protocol*, http://web.mit.edu/kerberos/www/, 2004

[7] Microsoft, *Crypto API Functions*, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnw98bk/html/thecryptoapifunctions.asp, 1998

[8] National Institute of Standards and Technology. Security *Requirements for Cryptographic Modules* (FIPS PUB 140-2), Washington, DC, 2001.

[9] Schneier, B. *Applied Cryptography, Protocols, Algorithms, and Source code in C, (2nd Edition)*, Wiley & Son, New York, 1996.

[10] Slemko, M., *Microsoft Passport to Trouble*, http://alive.znep.com/~marcs/passport/, 2001

[11] Stallings, W. *Cryptography and Network Security: Principles and Practices, (3rd Edition)*, Prentice-Hall, Upper Saddle River, NJ, 2003.

[12] Sun Microsystems, *Java Cryptography Architecture, API Specification and Reference* http://java.sun.com/j2se/1.4.1/docs/guide/security/CryptoSpec.html, 2002

[13] Sun Microsystems, *Cryptographic Accelerators*, http://www.sun.com/products/networking/sslaccel/, 2004


## 10. APPENDIX: MICROSOFT PASSPORT

The Microsoft Passport system is intended as the portal to all of Microsoft's new .NET services. The initial implementation of Passport provided a convenient method for securely storing a user's private information (address, credit card number, etc). That information could then be used when shopping with a Passport enabled web merchant. Figure 3 shows how this Passport system worked.

1) A user with a Passport account would be automatically logged onto the Passport server whenever he either logged onto his hotmail account or onto Windows XP.
2) The Passport server would download an encrypted cookie to the User's PC.
3) When the user shopped at a passport enabled web merchant, he would press a "Passport" button on a merchant web page.
4) The merchant web server would obtain the encrypted passport cookie from the user's PC.
5) The merchant web server would send the encrypted cookie to Microsoft's Passport server.
6) The Passport server verified that this request was coming from a legitimate merchant who had signed up for the passport service.
7) The Passport Server would then send the user's private information to the merchant web server, which would then display that information for the user to modify (if necessary).

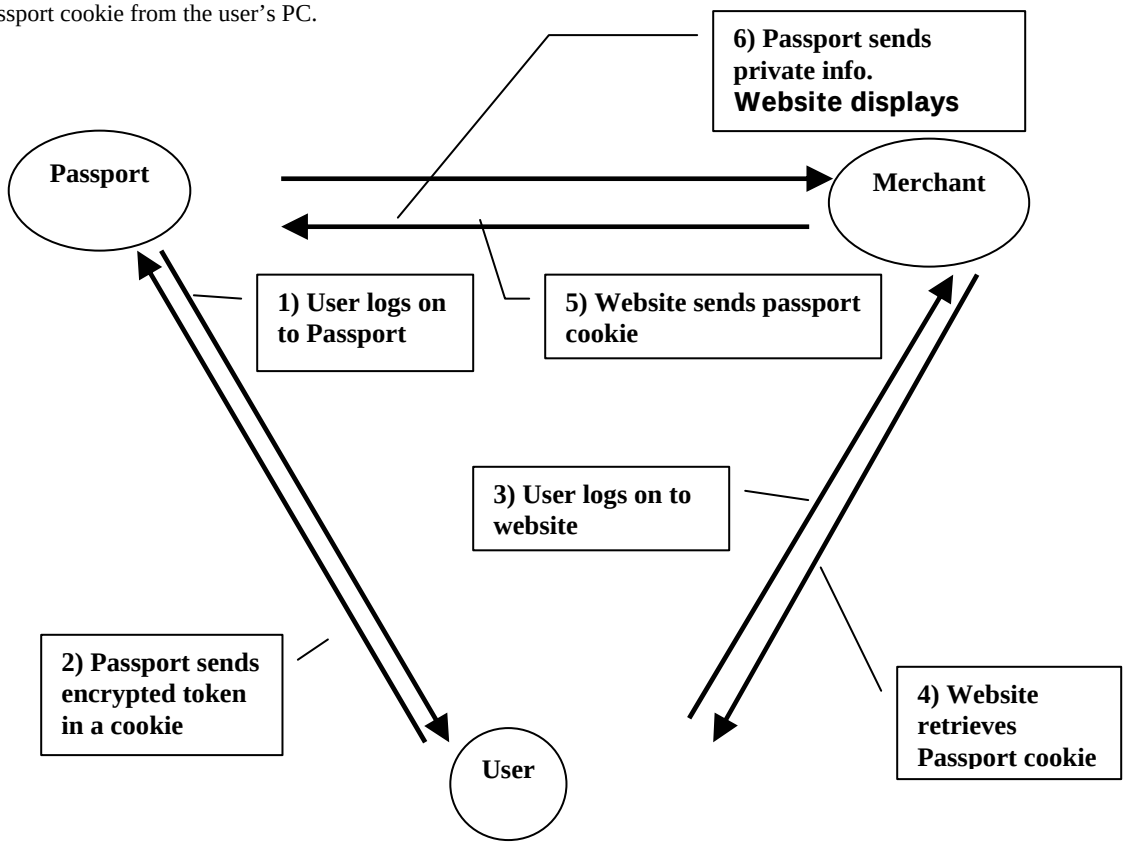



**Figure 3 Microsoft Passport Communication Protocol**

Figure 4 shows how this system was broken, using a Trojan horse and a form of cryptographic replay attack [10].

1) A hacker sends a specially formatted email to a user who has a passport account.
2) At some point, that user would logon to his hotmail account or Windows XP and be automatically logged onto the Passport server.
3) The Passport server would download an encrypted cookie to the User's PC.
4) When the user reads the email from the hacker, his email program will send the encrypted cookie to the hacker. This occurs because the email is a specially crafted html email that includes script commands to obtain the cookie.
5) The hacker then goes to a passport enabled web merchant and presses a "Passport" button on a merchant web page.
6) The merchant web server would obtain the encrypted passport cookie from the hacker's PC.
7) The merchant web server would send the encrypted cookie to the Passport server.
8) The Passport server verifies that this request was coming from a legitimate merchant who had signed up for the passport service.
9) The Passport Server would then send the user's private information to the merchant web server, which would then display that information for the hacker.

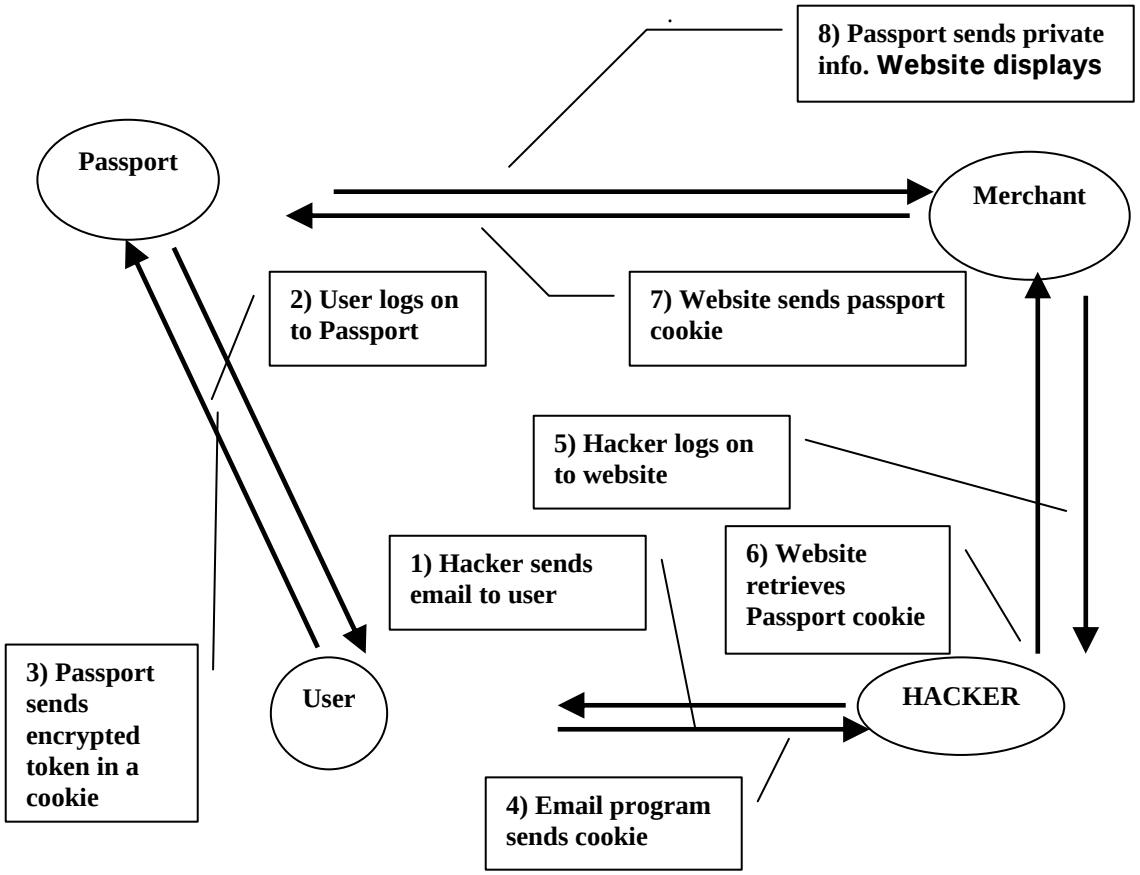



**Figure 4 Hacking Microsoft Passport**