Proceedings of the 12th Colloquium for Information Systems Security Education
University of Texas, Dallas
Dallas, TX June 2 - 4, 2008

Virtual Laboratories for Learning Real World Security

Tanya Zlateva, Leo Burstein, Anatoly Temkin, Andrew MacNeil, Lou Chitkushev, *Boston University*

***Abstract* – We present a laboratory module that follows an end-to-end security process pattern in securing real world applications. The overall goal is to relate theoretical concepts of cryptography and security protocols to implementation solutions and their use in the workplace. In a series of activities for installing, certifying and working with systems, each configuration decision and communication exchange is evaluated and discussed in the context of the theoretical knowledge acquired in our core courses in cryptography, network and software security, and network management and security. All systems are implemented as part of a virtual network environment thus reducing costs, and allowing the student easy access to different lab systems and the ability to play different roles and analyze security issues from the point of the systems manager, end user, cryptanalyst, or certification authority administrator.**

Index terms – security curriculum, virtual environment, securing network application, teaching

I. Introduction: The Theory—Workplace Chasm

The competing demands of theoretical knowledge and practical skills can be traced throughout the history of education and across disciplines—liberal arts vs. professional programs, formal theories vs. laboratory experiments, algorithmic knowledge vs. programming skills. However, in few fields is the contrast between theory and practice as stark and as important to reconcile as it is in information security: cryptographic algorithms draw on the most abstract branches of mathematics while their correct (or incorrect) application decides vital problems ranging from the confidentiality of the nation's critical infrastructure to the privacy of personal information. The importance of an interdisciplinary approach to information assurance and the need to integrate security topics throughout the curriculum have received early and well-deserved attention ([1-4]). In contrast to this little work was devoted to the educationally critical questions on how best to relate cryptographic concepts to laboratory experiments or to the security features of complex software applications. The lack of hands-on experience and/or the failure to explain clearly and in detail how the theory underlies practical solutions, how poor implementations can make strong cryptographic algorithms useless, or how inept configuration of standard security features in "out of the box" (commercial or open source) applications results in insecure systems, may leave the students with fragmented knowledge and insufficient practical skills. To address this problem we started developing laboratory modules that integrate knowledge from several security courses and include installing, configuring, and working with widely used commercial and open source applications in a virtual environment. For each laboratory activity we trace the relevant theoretical concepts and discuss their implications for securing the system. The choice of working with a complex application instead of developing targeted laboratory exercises for specific concepts is deliberate and important— students acquire practical knowledge of modern information security technologies and skills that are readily applicable in the workplace. They are better prepared to explain security problems, raise security awareness and build more secure infrastructures.

This paper presents our approach on the example of establishing secure communication between a server and client. The core of our security curriculum consists of courses in cryptography, network and software security, and network management and security. Our first lab series involves theoretical concepts from all three courses and illustrates these concepts on laboratory activities for building a virtual environment that includes an application server and client, a network protocol analyzer and a certification authority (CA) (Figure 1). Throughout the labs we emphasize three aspects: (i) the basic theoretical concepts such as confidentiality, security protocol; (ii) the technical implementation, such as setting up server with password protection, and securing it using Transport Layer Security (TLS) protocol; (iii) the human factors and organizational roles such as systems manager, end user, or hacker.

II. Learning about Systems and Roles in Virtual Laboratories

The laboratory exercise follows an end-to-end security process pattern, from identifying problems to securing systems, and use practical examples that are typical in today's Internet age. A realistic reflection of even a simple network environment requires setting up several systems, exploring the communications between the systems and the different roles of the participants. We chose to implement all systems using server virtualization technology that offers several important benefits: first, the virtual environment with all its components is available from a single computer leading to substantial savings in hardware. Second, the student can easily access every system of the virtual network environment from his/her own computer and explore security considerations from the prospective of the specific systems, e.g. server, client, network analyzer, certification authority, and also play different roles as they relate to these systems, e.g. system manager, user, cryptanalyst, security administrator. Depending on the size of the class, these roles can be interchangeably played by one or several students, and we might have several parallel environments to accommodate large classes. Third, the experiments are conducted in a controlled and secure environment. Last but not least the approach scales well as additional systems can be added with relative ease.

The scenarios we describe can be implemented on multiple platforms, using both open source and commercial software. In this paper we focus on Windows-based implementation; the actual selection of platforms will depend on instructor's and students' preferences. In larger classes, a combination of several platforms can be used, and lab exercises could be supplemented by students' research papers comparing various implementations.

We have successfully used virtualization technology to create basic system configurations that are used as starting points for building various systems in the laboratory. These building blocks can be then easily replicated, often without adding new hardware, to accommodate large courses and give instructors the flexibility to partition classes and implement different role-based scenarios. This sharply reduced and in many cases completely eliminated the overhead time students used to spend installing the systems and allowed them to focus their efforts on learning the material and meeting course objectives. Furthermore, virtualization has allowed us to "extend" the laboratory and make it possible for distance education students to partake in laboratory activities without the need to come on campus. We have a large student population (over 400 students) enrolled in online and blended classes and virtual labs are an important factor in increasing the quality of their educational experience.

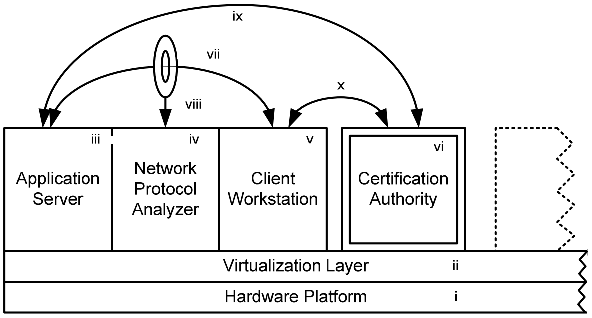


Figure 1. Virtual Laboratory: i) hardware platform; ii) virtualization layer; iii) application server; iv) network protocol analyzer; v) end-user workstation; vi) certification authority; vii) client-server communications; viii) eavesdropping; ix) server certificate processing; x) client certificate processing

III. Securing Commercial Applications—Theory and Implementation

A. Ensuring Fundamental Security Properties

In the first lab segment we explore the level of security. For this implementation we use Microsoft Windows Server 2003 R2 and students first install an application server (Internet Information Services, IIS, Figure 1.iii) with a simple home page that will simulate an enterprise application. All protocols we explore in this lab are platform-independent and the same laboratory configuration can be achieved using alternative platforms, e.g. Linux-based Apache server and/or Firefox browser on the client side. However, while the overall lab scenario remains the same, the setup instructions are specific to a particular operating environment. In the next step, we configure basic username/password authentication. At first glance this appears to be a reasonable choice for moderate security requirements (as would be the case for most browsers). This choice seems also to be supported by theory. The fundamental concepts of confidentiality, authentication, integrity, availability, non-repudiation are discussed in all three security core classes and students have learned that username and password is the simplest authentication scheme. Typically students have no problem understanding that authentication is achieved by "what you know, what you have, and what you are" and have been made aware of the vulnerability of passwords to eavesdropping. However, most are unaware of how weak (or rather nonexistent) this basic authentication scheme can be without adding additional protection layers. Despite the fair warning that passwords are sent in clear text, the possibility of eavesdropping at the right place and time appears remote, too complicated, or not worth the trouble. We show that eavesdropping is simple, no trouble, and thus not remote.

The second system students install is an open source network protocol analyzer (Figure 1.iv). We use Wireshark that is a free download from http://www.wireshark.org. Similar to the web browser, the sniffer is implemented on a virtual system. The end-user browser (Figure 1.v) can also be setup as a separate virtual system, or students can just use their computers or classroom computers, as this step does not require installing any additional software.

Once communication between client and server is initiated (Figure 1.vii), the protocol analyzer intercepts the username and password that are sent in clear text over a virtual network connecting the lab systems. Students learn basics of using a network protocol analyzer, become familiar with the data format of the HTTP protocol and where to locate the user's credentials. Independently of security considerations, this is useful practical knowledge that might help them in future software development and systems troubleshooting.

Microsoft IIS web server, as well as all other widely used application platforms, offer a mechanism to protect client-server communications that is known as Transport Layer Security, or TLS protocol. This security protocol uses public key cryptography to identify the server (and optionally the client) and then establish a secure communications channel that protects the information being exchanged, including username/password credentials. The next several steps cover the basic steps involved in TLS setup: building a certification authority, generating key pairs, creating and sending certificate requests, issuing and delivering certificates, and negotiating a secure channel to exchange data.

B. The Interplay of Secret and Public Key Cryptography in Security Protocols

As noted by Ross Anderson "if security engineering has a unifying theme, it is the study of security protocols" [7]. Indeed, this theme resonates across the core courses in cryptography, network and software security, and network management and security, and continues with more depth in the specialized high-level electives. Thus, it is extremely important to ground the student early on in the theoretical and applied aspects of security protocols. This part of the lab focuses on relating cryptographic theory to the design of the data exchange protocols and key management strategies. These topics are covered in all three basic courses with a different emphasis and detail, and following different textbooks ([5-7]). We first briefly summarize the material covered in the three courses that provide the prerequisite knowledge for the lab.

The algorithmic foundation is provided in the cryptography course. A rigorous treatment of hash functions begins with the definition of one-way functions, followed by the example of discrete exponentiation, and detailed review of the MD5 and SHA-1. Group theory fundamentals provide the entry for the discussion of secret and public encryption and their use in protocols for authentication and more generally for secure communications. Special care is taken to compare and contrast different cryptographic algorithms in terms of their vulnerability to attacks as well as computational aspects, such as implementation pitfalls and overhead.

The network and software security and the network management and security courses shift the emphasis from the specific cryptographic algorithms to applying a variety of cryptographic tools to achieve a security task under different threats and security models. The basic security models—Bell-LaPadula, Biba, Chinese Wall — are discussed in relation with confidentiality, integrity, and hybrid policies. Network security topics include a broad range of security mechanisms, more specifically authentication and confidentiality with symmetric and asymmetric encryption methods, IPsec, SSL/TLS, e-mail security, PGP. Application security topics are discussed using examples of Java and .NET security architectures. The important problem of key management in secret and public encryption implementations, along with the pros and cons of key distribution centers and certification authorities, is discussed in all three core courses.

For this part of the lab we chose the typical and practical task of setting up secure communications between web servers and browser clients, because the underlying principles cut across network and application level security and involve interesting cryptography. The configuration decisions hinge upon important theoretical and computational trade-offs, such as choosing the key length and the hash algorithm, understanding digital certificates, how to request them from a certification authority, how to use them, and how to check their validity. The lab concludes with the establishment of secure communications over TLS that provides the basis for discussing the pros and cons, and the typical uses of public and secret encryption.

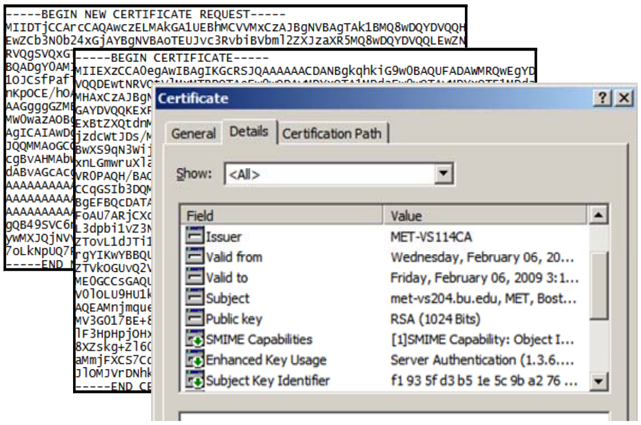
We view the need of complementing practical knowledge with the underlying theoretical principles and the pitfalls of setting up application security of key importance for the successful security professional. While very few will design new cryptographic algorithms, virtually all will use authentication protocols, and some might need to develop their own security protocols. In the following sections we outline the laboratory activities and related discussions at each step of the lab.

C. Cryptographic Keys Generation and Certification

In this step, the student uses cryptographic services to generate a key pair for the application server and create a certificate request. To provide certification services for the rest of the labs, students build their own certification authority on a separate virtual machine.

In IIS key generation is part of the process of requesting a certificate from the CA. The student, in the role of the web server manager, chooses the key length, provides the name of the organization for which the certificate is required and other necessary parameters, and then submits a certificate request. There are several important aspects that must be addressed at this point: first, the trade-off between computational overhead and encryption strength depending on the length of the key as discussed in the courses from a theoretical point of view; second, key management as a critical success factor in implementing enterprise solutions, and the role and main components of Public Key Infrastructures (PKI). More narrowly at this stage it is important to point out what key is sent and what exactly is requested—it must be made clear that the server sends its public key to the CA and requests that the CA certifies this key as belonging to the server, while server's private key never leaves the server. If the certificate request meets the CA policy requirements (in our case, students themselves learn to play the role of CA administrators), CA will issue a digital certificate and make it available for download by students playing the application server manager roles. This opens the question "How does the CA administrator know the server is who it claims it is?", and introduces the third discussion topic—non-technical breaking points and importance of securing out-of-bound communications.

In addition to fundamental discussion questions outlined above students acquire practical experience in setting up a CA and learning the basic steps necessary to process certificate requests. The mystery of "digital certificates" is eliminated by looking at the format of certificate and certificate requests (Figure 2).



```
-----BEGIN NEW CERTIFICATE REQUEST-----
MIIDTjCCArcCAQAwczELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAk1BMQ8wDQYDVQQH
EwZCb3N0b24xGjAYBgNVBAoTEUJvc3RvbiBVbml2ZXJzaXR5MQ8wDQYDVQQLEwZN
RVQgSVQxG
BQADgY0AM
1oJCsfPaf
nKpOCE/hO
AAGgggGZM
MW0wazAOB
AgICAIAwD
JQQMMAoGC
cgBvAHMAb
dABvAGCAC
AAAAAAAAA
AAAAAAAAA
AAAAAAAAA
gQB49SVC6
ywMXJQjNV
7oLkNpUQ7
-----END
```

```
-----BEGIN CERTIFICATE-----
MIIExzCCA0egAwIBAgIKGcRSJQAAAAAACDANBgkqhkiG9w0BAQUFADAWMRQwEgYD
VQQDEwtNRVQ
MHAxCzAJBgN
GAYDVQQKExF
ExBtZXQtdnM
jzdcWtJDs/M
BwXS9qN3Wij
xnLGmwruXla
VR0PAQH/BAQ
CCqGSIb3DQM
BgEFBQcDATA
FoAU7ARjCXQ
L3dpbi1vZ3N
ZTovL1dJTi1
rgYIKwYBBQU
ZTVkOGUvQ2V
ME0GCCsGAQU
V0lOLU9HU1k
AQEAMnjmque
MV3G017BE+8
lF3HpHpjOHx
8XZskg+Z16C
aMmjFXCS7Cc
JlOMJVrDNhk
-----END CE
```

Certificate

General | Details | Certification Path

Show: <All>

Field	Value
Issuer	MET-VS114CA
Valid from	Wednesday, February 06, 20...
Valid to	Friday, February 06, 2009 3:1...
Subject	met-vs204.bu.edu, MET, Bost...
Public key	RSA (1024 Bits)
SMIME Capabilities	[1]SMIME Capability: Object I...
Enhanced Key Usage	Server Authentication (1.3.6....
Subject Key Identifier	f1 93 5f d3 b5 1e 5c 9b a2 76 ...

Figure 2. Certificate request, issued certificate, and its human-readable representation

D. Secure Communication

After receiving the certificate, the web server manager installs it and configures the server to accept only secure https requests. Students discuss the "conversation" between the browser and the server that is used in establishing a secure communications channel.

The "end user" uses https to access the server, and gets a security warning because his browser does not recognize the CA signature. Students learn how to configure trusts, how to check web site identities, risks of overriding security warnings, etc. After students add the CA certificate to trusted roots, they can transparently establish secure web sessions. Students take a trace of client-server communications to ensure the traffic is encrypted, and usernames/passwords are not clearly present.

Students discuss the interaction between symmetric and asymmetric keys used at different steps of negotiating a TLS session, pluses and minuses of symmetric and asymmetric algorithms from the practical perspective, use of TLS in their daily lives – online banking, online stores, etc. They discuss how digital certificates that they just learned can help to protect against phishing and other scams.

The next step is to establish bidirectional authentication. In the previous steps, students learned how to establish and verify the identity of a web site. Now students simulating end users get their own digital certificates. They use the same CA established earlier. The "web site manager" configures the web site to require client-based authentication. Authorization rules are configured to provide access to the web site to users with digital certificates signed by the lab CA. When trying to access the site, students receive a request to present their digital certificates. They learn more about different certificate stores, differences between different types of certificates (server authentication, client authentication, email signing, etc.). Students also review the theory behind digital signatures they learned in the theoretical part of the course (digests, signing, etc.).

IV. Conclusion and Future Work

We presented a first laboratory module that follows an end-to-end security process pattern in securing a real world application. The advantage of this approach as compared to laboratory exercises tailored to specific algorithms or protocols is that it binds the theory to practical knowledge about widely used applications and teaches the necessary skills for working with these applications. All laboratory activities are conducted in a virtual network environment that is secure, reduces costs, and allows students to easily assume different roles—security managers at server side or at the certification authority, end users, hackers—and gain a keen appreciation of the different points of view.

For our first implementation, we used a virtualization layer based on Microsoft Virtual Server 2005 R2 SP1 (available as a free download at *http://www.microsoft.com/virtualserver/*), running on top of Windows 2003 Server. Open source Wireshark protocol analyzers were installed on our computing lab workstations and students were analyzing traffic between their workstations and virtual application servers.
A detailed laboratory manual was developed as a research project by one of our scholarship students. Reflecting on his experience, he wrote that "... seeing the actual risk of an insecure system and then learning actual ways of how we can mitigate this vulnerability has had a lasting impact on my understanding of computer security". The complete laboratory module was piloted in student research seminars and, starting in the Fall 2008 semester, will become a standard part of the basic network and software security courses of the MS programs in computer science, computer information systems, and telecommunication. The virtual laboratory approach was also used in a large online database security course and was a major factor for the enthusiastic student feedback, as seen from the following comments:
"Thanks for a great course! ...I liked the way you challenged us to consider a broader context for the subjects we were learning, or to apply what we had learned to areas beyond what was presented in the book and lectures."
"This is the final course for me in this program of study. What a great way to finish up! Thanks!"
"It's a great course, clearly a useful resource for current and future DB security opportunities."
"This was one of the best courses in the MSCIS program. The labs promoted hand-on learning and reinforced the readings. Great course design..."
"I have really appreciated the way that you have encouraged everyone to look at things in a new way and learn more than what was ever in the scope of this course."
"I live and breathe security everyday at work and will take your comments with me."
"Thanks for a creating a great class. The course really rounded out my knowledge in an area I was never comfortable in... I was able to understand the technology and ask some insightful security questions which I wouldn't have been able to do seven weeks ago. It was an immediate return on my investment..."
"Thanks for the great class. I have really enjoyed this and almost regret that it has to come to an end."

We plan to expand this approach and develop a series of laboratory modules to address complex security problems in the context of business applications using commercial and open source software, and integrate knowledge that is taught in courses across the security curriculum. More specifically, we are currently working on scaling our approach and developing best practices to accommodate larger online classes, as well as on introducing several advanced (and sometimes forgotten) topics such as certificate revocation lists and access control through mapping of digital certificates to actual user accounts. Another direction for future work is the focused assessment of learning outcomes. We are aware that while students enjoy seeing the usefulness of theory in real-world applications this is not enough for concluding if and how learning has improved. To assess how the added laboratory modules affect learning in general, and more specifically what knowledge domains (if any) benefit most from the hands-on component we are developing pre- and post-laboratory tests that will address theoretical as well as applied aspects included in the laboratory tasks.[1]

V. References

[1] Irvine, Cynthia E., Chin, Shu-Kai, and Frinke, Deborah. "Integrating Security into the Curriculum", IEEE Computer, Vol 31, No 12, pp.25-30, 1998.

[2] Zlateva, T, V. Kanabar, A. Temkin, L. Chitkushev, S. Kalathur. Integrated Curricula for Computer and Network Security Education, Proceedings of the Colloquium for Information Systems Security Education, Society for

[1] We thank our anonymous reviewers for pointing out the aspects of implementation and assessment that needed clarification and helping us to improve this paper.

Advancing Information Assurance and Infrastructure Protection, Washington, D.C., June 2003.

[3] Dodge, Ron, Dan Ragsdale**: "**Technology Education at the US Military Academy", IEEE Security and Privacy, vol. 3, issue 2, pp. 49 – 53, March 2005.

[4] Bishop, M. "Teaching Context in Information Security," Journal on Educational Resources in Computing 6(3) article #3, September 2006.

[5] Paul Garrett: Making, Breaking Codes: An Introduction to Cryptology. Prentice Hall, 2001.

[6] Charlie Kaufman, Radia Perlman and Mike Speciner, Network Security - Private Communication in a Public World, 2nd ed, Prentice-Hall, 2002.

[7] Ross J. Anderson: Security Engineering: a Guide to Building Dependable Distributed Systems. Wiley Computer Publishing, 2001.