Network Security Application Layer

Target Course

Networks

Learning Goals

A student shall be able to:

1. Describe foundational security concepts in securing networks and systems.
2. Describe security design principles and identify security issues associated with common threats and attacks.
3. Apply principles of secure design and defensive programming techniques when developing software.

IAS Outcomes

IAS Knowledge Topic	Outcome
Cryptography	4. Explain how public key infrastructure supports digital signing and encryption and discuss the limitations/vulnerabilities. [Familiarity] 5. Use cryptographic primitives and describe their basic properties. [Usage]
Network Security	2. Describe the architecture for public and private key cryptography and how public key infrastructure (PKI) supports network security. [Familiarity] 3. Describe virtues and limitations of security technologies at each layer of the network stack. [Familiarity] 4. Identify the appropriate defense mechanism(s) and its limitations given a network threat. [Familiarity]

Dependencies

- Cover after the ***Network Security Concepts*** module.

Summary

Describe how the application layer may be used to support the security goals of CIA and the fundamental concepts of assurance, authentication, anonymity, and non-repudiation.

Estimated Time

This module took approximately one lecture hour to cover.

Materials

How does this layer affect the security goal of confidentiality?

- Use cryptography on any application data being transmitted between networked devices. Refer to the Network Security Concepts module for more information about cryptography.
- Adhere to as many of the security principles as possible when designing and implementing a system. These include the ten principles first published by Saltzer and Schroeder in 1975 [3] and the five additional principles published by McGraw in 2013 [4]. See the *Network Security Concepts* module for details.

How does this layer affect the security goal of integrity?

- Use a cryptographic hash function on any application data being transmitted between networked devices. The sender can compute the hash of the plaintext and encrypt the plaintext. It would then send both the ciphertext and the hash values. The receiver would decrypt the data and compute the hash, and then compare this to the hash value received. Refer to the Network Security Concepts module for more information about cryptography.

- Adhere to as many of the security principles as possible when designing and implementing a system. These include the ten principles first published by Saltzer and Schroeder in 1975 [3] and the five additional principles published by McGraw in 2013 [4]. See the *Network Security Concepts* module for details.

How does this layer affect the security goal of availability?

- Validation of data transmitted via an application protocol is critical. While this may not prevent a DDOS attack from being successful, it should reduce the work being performed by the service in trying to respond to the flood of invalid requests.
- Adhere to as many of the security principles as possible when designing and implementing a system. These include the ten principles first published by Saltzer and Schroeder in 1975 [3] and the five additional principles published by McGraw in 2013 [4]. See the *Network Security Concepts* module for details.

How does this layer affect the fundamental security concept of assurance?

- The design of an application layer protocol should address relevant policies, permissions, and protections.
- Adhere to as many of the security principles as possible when designing and implementing a system. These include the ten principles first published by Saltzer and Schroeder in 1975 [3] and the five additional principles published by McGraw in 2013 [4]. See the *Network Security Concepts* module for details.
    - Of particular importance is the principle of *psychological acceptability*, which says that a HCI should be designed to match the protection mechanism being used to the user's mental image of their protection goals. This should increase a users' trust in the system.

How does this layer affect the fundamental security concept of authenticity?

- Use a digital signature and/or digital certificate if public key encryption is available to use. Otherwise, include in the application data being transmitted data that allows the destination to authenticate the sender. Refer to the Network Security Concepts module for more information about cryptography.
- Adhere to as many of the security principles as possible when designing and implementing a system. These include the ten principles first published by Saltzer and Schroeder in 1975 [3] and the five additional principles published by McGraw in 2013 [4]. See the *Network Security Concepts* module for details.

How does this layer affect the fundamental security concept of anonymity?

- Adhere to as many of the security principles as possible when designing and implementing a system. These include the ten principles first published by Saltzer and Schroeder in 1975 [3] and the five additional principles published by McGraw in 2013 [4]. See the *Network Security Concepts* module for details.
    - Of particular importance is the principle of *promote privacy*, which says that the types of personal information being collected from a user should be carefully analyzed. Specifically, does the system really need the information being requested? Should the personal information be encrypted? Does this data really need to be persistently stored?

How does this layer affect the fundamental security concept of non-repudiation?

- Adhere to as many of the security principles as possible when designing and implementing a system. These include the ten principles first published by Saltzer and Schroeder in 1975 [3] and the five additional principles published by McGraw in 2013 [4]. See the *Network Security Concepts* module for details.
    - Of particular importance is the principle of *compromise recording*, which says to record the details of user and system actions so that an intrusion can be detected.

Assessment Methods

Below are questions that have been used on quizzes and exams.

Why is DNS one of the most insecure protocols on the Internet?

a. This protocol assumes trust between DNS servers.

b. This protocol authenticates the requests but not the responses.

c. All of the above.

d. None of the above.

Answer: a. This protocol assumes trust between DNS servers.

HTTP will do authentication of the server but not the client.

a. True.

b. False.

Answer: b. False.

HTTPS uses public key cryptography to establish a secret key between the client and server.

a. True.

b. False.

Answer: a. True.

Why is the application layer so important with regarding to meeting security goals and applying security concepts?

a. Because the other layers satisfy many but not all of the goals/concepts.

b. Because the other layers were developed by someone else, and so we should be cautious about relying on these to satisfy our security goals/concepts.

c. All of the above.

d. None of the above.

Answer: d. None of the above.

Explain why the security goal of confidentiality is so important when developing a distributed application?

Answer: If the security goal of confidentiality is not considered when developing a distributed application, then an individual may be able to:

- *Gain access to persistent data that is in plaintext form (i.e., has not been encrypted).*
- *Gain access to data in transit that is in plaintext form (i.e., has not been encrypted).*
- *Gain access to data that they are not authorized to view, update, or delete.*

Explain why the security goal of integrity is so important when developing a distributed application?

Answer: If the security goal of integrity is not considered when developing a distributed application, then an individual may be able to:

- *Update or delete data even though they do not have the authority to do so.*

This may lead to a lack of trust in the distributed application.

Explain why the security goal of availability is so important when developing a distributed application?

Answer: If the security goal of availability is not considered when developing a distributed application, then an individual may be able to:

- *Cause a denial-of-service by sending many requests/packets to a service thus keeping it too busy to respond to valid requests.*

This would prevent information from being delivered in response to a valid request in a timely fashion.

Explain why the security goal of non-repudiation is so important when developing a distributed application?

Answer: If the security goal of non-repudiation is not considered when developing a distributed application, then an individual may be able to:

- *Alter some data (e.g., their salary) without any proof that this individual altered data they were not supposed to have access to.*

This would make attribution of an attack much harder to determine.

As a designer of software applications, what types of questions should you be asking and discussing to help promote privacy of data?

Answer: Some sample questions:

- *Do we really need to obtain and either store or transmit this private data?*
- *Which of this data may be used maliciously to identify our users/customers?*
- *Which of this data should we encrypt while it is persistently stored?*
- *Which of this data should we encrypt while it is in transit?*

An application-layer protocol design that was part of a semester-long project. In the later stages of this project, security mechanisms are added to give students experience using various cryptographic tools found in the Java API.

What follows is the summary page used to introduce the students to this assignment. More details for this assignment may be obtained by requesting the *App-Layer Security Mechanisms* assignment from voorhedp “at” Lemoyne “dot” edu.

1. Modify the Python client and the Java server source code files to provide security mechanisms described below. You may use the instructor's solution for both the client and server as your starting point.

2. Implement one of the security mechanisms described in 2.a or 2.b.

   a. Implement the SSL/TLS protocols between the Python-based client and Java-based server. See the **SSL/TLS Protocols** section for more details. When you choose this option, this is the first thing your client and server should do (i.e., establish an SSL/TLS connection). Once the secure channel is established, the client and server should follow the **Authentication Protocol**. A student may earn up to 15 bonus points if they document their attempts at getting this to work.

   b. Implement cryptography as described in the **Synchronous Cryptography** section. When you choose this option, all messages sent between your client and server should be encrypted. This includes messages that do authentication.

3. Implement authentication as described in the **Authentication Protocol** section.

4. As a bonus, implement logging as described in the **Compromise Recording Log** section. Implementing logging on the client is worth up to 10 bonus points. Implementing logging on the server is worth up to 10 bonus points.

- ese classes is in plaintext.

The table below identifies the security mechanisms used for steps 2 through 4.

Security Mechanism	Step 2a	Step 2b	Step 3	Step 4
Establish SSL/TLS connection	Yes			
Public key cryptography (RSA)			Yes	
Synchronous cryptography (AES)		Yes		
Cryptographic hash function (SHA-256)			Yes	
Write log file				Yes

References

[1] M.T. Goodrich & R. Tamassia, (2011). *Introduction to Computer Security*. Addison Wesley.

[2] R. Anderson, (2008). *Security Engineering, Second Edition*. Wiley.

[3] J.H. Saltzer & M.D. Schroeder, (1975). The protection of information in computer systems. *Proceedings of the IEEE*, 63(9), 1278-1308.

[4] G. McGraw, (2013). Thirteen principles to ensure enterprise system security. Retrieved on July 28, 2015 from searchsecurity.techtarget.com/opinion/Thirteen-principles-to-ensure-enterprise-system-security.