(IJAER) 2012, Vol. No. 4, Issue No. V, November ISSN: 2231-5152

AUTOMATIC PROTOCOL CREATION FOR INFORMATION SECURITY SYSTEM

Mr. Arjun Kumar
ABES Engineering College, Ghaziabad
Master of Computer Application

ABSTRACT

Now a days, security is very big problem in life and IT sector, so this paper fully based on automatic protocol creation, and security protocols for IT sector. Automatic Protocol creation, APC for short, is a mechanism to generate security protocols automatically for any type of transaction. Advantage of the Automatic Protocol Creation (APC) is fully secure transaction without any disturbance approach over the current protocol design process is that, it is fully automatic. The designer inputs the properties and system requirement's which result in a security protocol or output - a better process than creating the security protocol manually.

INTRODUCTION

This paper will attempt to describe automatic protocol Creation, and security protocols. Automatic Protocol Creation, APC for short, is a mechanism to generate security protocols automatically. This is accomplished by having the designer or engineer input a set of security system requirements and properties that dynamically produces a security protocol that best meets the criteria. The system requirements for input are defined as a metric function, which defines the cost or overhead of the protocol primitives, which defines an ordering over protocols with respect to the metric function. The advantage of the Automatic Protocol Creation (APC) approach over the current protocol design process is that, it is fully automatic. The designer inputs the properties and system requirement's which result in a security protocol or output. This is by far, a better process than creating the security protocol manually. The protocols creation by APC has a higher level of confidence. This high level of confidence is a result of being able to verify with a powerful protocol analyzer. Another advantage of APC is that since with respect to the order of increasing cost on the Metric Function, APC searches through the protocol space and generates correct protocols with minimal cost that are in line with the systemrequirements.

SECURITY PROTOCOLS

Security Protocols is very essential scheme of ecommerce and the Internet. Every day new hacker's compromises and viruses threat the ability to securely and effectively transact data between "authorized users" on the Internet. The role of a Security Protocol is to utilize cryptographic building blocks to achieve security goals such as authentication, reliable, accuracy and integrity. Current security protocol design process is following reasons: It's Error-prone. Security protocols should be intricate because hackers are powerful. Manually designed protocols are flawed because they contain undocumented assumptions, which is a result of the lack of formalism and mechanical assistance. The protocol designer lacks the expertise and experience and is more than likely to develop a non-efficient protocol that is flawed fundamentally. The protocol creation after receiving the required inputs it creates a candidate security protocol, which satisfies the system requirements. In the final stage a protocol screener analyzes the candidate protocols, ignores the flawed protocols, and creates the correct protocols that address the desired security properties. The benefits of this approach are that it provides the following:

Automatic, the designer specifies the security parameters and properties but the remaining process is automatic, provides a High Confidence level. There are no hidden assumptions as is the case in the manual protocol development process. The protocol screener is powerful enough to generate a proof if the protocol is correct or a counterexample. Thus, providing further Confidence in the process. High Quality, the user defined requirements include a metric function which specifies cost overhead of a protocol. Flexibility, this mechanism works for different security properties, system requirements, and attackermodels.

PROTOCOL CREATION

So many problems come in mind how automatic protocol create, what is role of protocol create. The Protocol Creation's function is to generate candidate protocols that satisfy the particular system requirements. When we observe closely the protocol space that the protocol creation works with, one finds that that space is virtually infinite. This poses another challenge, how do we limit the number of potential protocol candidates without omitting any potential optimal protocols. The answer to this question lies in a process called *iterative deepening.* This process, plain and simply put, is a search algorithm. The way this algorithm works is that a cost threshold of protocols is set in each iteration. Then a search is done in the protocol space to create all the protocols below the given threshold. Next, after the protocols are sorted by their costs, the protocol screener tests them. If a protocol satisfies the desired properties (Which means that its cost is minimal) the creation process can stop. Otherwise, we increase the cost threshold and generate more protocols. To also aide in the process, a reduction technique is used to prune invalid candidate protocols early before they are passed on to the Protocol Screener. Many of the created protocols include severe security flaws, which can be detected by a verification algorithm. A pruning algorithm is used to discard most severely flawed protocols. The Protocol Screener uses a verificationcondition.

PROTOCOL SCREENER

The role of the Protocol Screener is given a candidate protocol, the screener must examine the protocol and tell whether it's verifiable or not. The protocol screener is reliable when it claims that a protocol satisfies certain security properties. Since the protocol generator produces thousands of protocols, the protocol screener is required to be very efficient in its task to find the optimal protocol in a reasonable amount of time. So the next logical question would be, how does the Protocol Screener handle the task of verification of the protocols it receives from the generator? To answer this question we must look at a few verificationtechniques.

Basically there are **two types of verification techniques.**

(i) Semiautomatic protocolanalysis

(ii) Automatic.

Semiautomatic protocol analysis tools are NRL Analyzer [Mea94], the Interrogator Model [Mil95], FDR [Low96], and Brutus [CJM98]. Althena however, is an automatic protocol analysis tool that is most preferred automatic protocol analyzer because of the following reasons; Althena has the ability to analyze protocol executions that have any arbitrary configurations. Many existing automatic analyzer tools can only reason about finite state space. When Althena terminates, it proves that a protocol satisfies its specified properties under any arbitrary protocol configuration, or it demonstrates a counterexample if the property does not hold. Althena can exploit state space reduction techniques, which as a result provides a highly reduced state space.

LAY OUT OF PROTOCOL

A protocol represents the sequence of actions of two communicating parties. The actions include sending and receiving messages. These messages are defined by the grammar listed below, and can easily be extended as needed. This also helps support the argument of flexibility in APC.

Message ::= Atomic |Encrypted |Concatenated
Atomic ::= name of protocol |Nonce | Any Key
Encrypted ::= (Message, Any Key)
Any Key ::=PublicKey | PrivateKey| SymmetricKey
Concatenated ::= Message List
Message List ::= Message | Message, Message List

A tree can also represent each Message, with the atomic messages as leaves and operations as intermediate nodes. In the figure below we illustrate an example for the message ***A, B, {A, B} kb.*** The *depth* of a message is defined as the depth of the tree representing the message.

In the example below the message depth is 4.

Notation

A, B are the principals
NA is a nonce generated by A
KA denotes A’s public key
KA-1 denotes A’s private key

Case may be arises

For the purposes of keep this paper brief we have elected just to give a brief summary of the case study found . However, I recommend further reading of this case study, because helps to simplify the complexities of the process.

1. Assumptions
   - (a) Message components aretyped
   - (b) No redundant message components in theconcatenation
   - (c)No initial keys are sent in amessage.
   - (d) The initiator’s name needs to be in the first message in a format understandable to the responder.
   - (e) We don’t consider permutations of the message components of a concatenated message.

2. A pruning Algorithm is developed for each security property, which prunes the majority of the protocols.

3. For impersonation attempts, we use two intruders to attack each protocol. The Intruder I1 tries to impersonate the initiator. A, and the other intruder Ir , tries to impersonate the responderB.

4. If the protocol screener outputs a flawed protocol, the automatic protocol generation is not trustworthy. The screener has to be efficient because the generator could generate thousands of protocols.

5. A simple linear metric function is used in the experiment. Each operation has a unit-cost. The cost value of a protocol is the sum of the costs of all the protocol operations andcomponents.

Example of Symmetric-Key mutual authentication protocols:
A -> B: NA, A
B -> A: { NA, NB, A} KAB
A -> B: NB
A -> B: NA, A
B -> A: { NA, NB, B} KAB
A -> B: NB
Example of ISO Symmetric-Key three-pass Mutual Authentication
Protocol:
A ->B: NA, A

B ->A: { NA, NB, B} KAB
A ->B: { NA, NB} KAB
Example of Asymmetric-key mutual authentication protocols:
(The following protocol is the same as the fixed version of Needham-Schroeder protocol[Low96])
A -> B: { NA, A} KB
B -> A: { NA, NB, B} KA
A -> B: NB

Stop Discussion but so many question in mind

With a user-defined specification of security properties and system requirements, including a system metric function, APC generates minimal protocols that satisfy the specified security properties and system requirements, minimal with respect to the metric function. This strategy is a significant improvement over the current protocol design process, because it is more reliable, efficient, and produces protocols that are comparable to the given system requirements.

REFERENCES

[Low96] G. Lowe. Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In Tools and Algorithms for the Construction and Analysis of Systems, volume 1055 of Lecture Notes in Computer Science, pages 147-166. Springer-Verlag, 1996.

[Tygar01]
Atomicity versus Anonymity: Distributed Transactions for Electronic Commerce (J.D.Tygar, 1998) [New] 06/05/01

[BAN89]
M. Burrows, M. Abadi, and R. Needham. A logic of authentication. Technical Report 39, DEC Systems Research Center, February 1989.

[CGP99]
Edmund Clarke, OrnaGrumberg, and DoronPeled.Model Checking.MIT Press, 1999.**[CJM98]**
E.M. Clarke, S. Jha, and W. Marrero.Using state space exploration and a natural deduction style message derivation engine to verify security protocols.In Proceedings of the IFIP Working Conference on Programming Concepts and Methods (PROCOMET), 1998.**[Int93]**
International Standards Organization. Information Technology - Security techniques Entity Authentication Mechanisms Part 3: Entity authentication using symmetric techniques, 1993. ISO/IEC 9798.

[Low96]
G. Lowe. Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In Tools and Algorithms for the Construction and Analysis of Systems, volume 1055 of Lecture Notes in Computer Science, pages 147-166. Springer-Verlag, 1996.

[Low97]
G. Lowe. A hierarchy of authentication specifications. In Proceedings of the 1997 IEEE Computer Society Symposium on Research in Security and Privacy, pages 31-43, 1997.

[Mea94]
C. Meadows. A model of computation for the NRL protocol analyzer.In Proceedings of the 1994 Computer Security Foundations Workshop. IEEE Computer Society Press, June 1994.

[Mil95]
J. Millen. The Interrogator model. In Proceedings of the 1995 IEEE Symposium on Security and Privacy, pages 251-260. IEEE Computer Society Press, 1995.

[WL93]
T. Y. C. Woo and S. S. Lam. A semantic model for authentication protocols.In Proceedings of the IEEE Symposium on Research in Security and Privacy, 1993.