Authentication, Integrity And Non-Repudiation

Authentication is one of the main purposes of cryptography and includes several aspects:

- authentication of parties;
- authentication of a message source;
- transmitted data integrity check;
- non-repudiation.

Authentication of parties (identification) includes methods that enable one of the parties, called a checking party, to verify in real time mode whether really authorized users participate in the communications session. Principal difference between authentication of parties and authentication of a message source lies in the fact that authentication of a message source takes place independently of the message creation time. On the contrary, authentication of parties takes place at a specific point of time during execution of the authentication protocol. If continuous verification is required, additional methods are used.

Weak authentication, strong authentication and zero-knowledge protocols are recognized.

Weak Authentication Of Parties (Passwords)

Each user of a communications network is given a password (usually 6 to 10 characters long). In order to get an access to public resources of a computer network, the user enters his password and specifies the resource the user wants to access. In this case, user's knowledge of a confidential password indicates that he is authorized to establish communication. The system verifies whether the user with such password has an access to this given resource.

Various password schemes exist. They vary in password verification methods and the ways password information is stored in the system. There are fixed password schemes that use PIN codes, and schemes that use one-time passwords.

Strong Authentication Of Parties (Request-Response Schemes)

The concept of such authentication implies that one of the parties confirms its authority to participate in the communications session by demonstrating its knowledge of certain confidential information that has been provided to this user in advance, but does not disclose such information to the second party during execution of the identification protocol. Some schemes imply that the second party knows such confidential information, other schemes imply that it does not. This can be accomplished due to the fact that a request is a number selected at random and on a confidential basis in the beginning of the identification protocol, while a response depends on the confidential information and request. Therefore, a response contains no information about subsequent identifications, because requests are different every time.

There exist request-response schemes that are based on symmetrical algorithms, public key algorithms, and those that use a zero-knowledge concept.

Random and pseudorandom numbers, sequence numbers (serial number and counter value), as well as time stamps are used as a request. Each of them has its own advantages and disadvantages. Numbers used as a request must be random and equiprobable, while pseudorandom numbers must not be repetitive. When sequence numbers are used, each user must keep in its memory information about the sequence number of the last message received from any other user. Therefore, such authentication schemes are suitable for networks with a limited number of subscribers only. When time stamps are used, one of the parties enciphers a time stamp obtained from its clock and transmits it to the second party. The second party deciphers this stamp and compares it with its own stamp obtained from its clock. The received time stamp and own time stamp may not differ from one another more than by a previously specified value that depends on the maximum message passage time in the communications network. In such a case, continuous synchronization and clock confidentiality are required for the network subscribers.

Zero-Knowledge Protocols

Authentication protocols that belong to this class use asymmetrical algorithms that are different from digital signature algorithms and public key encryption algorithms. They are based on the same number theory problems as public key schemes (discrete logarithm, factorization of large numbers, etc.) and are similar to the request-response schemes, although they do not use block ciphers, sequence numbers and time stamps, but are built on interactive proof systems and use random numbers instead. In the request-response schemes, an enemy may obtain information about the confidential key, while in the schemes that use the zero-knowledge concept the parties provide proofs of statement validity without transmitting the statement itself in any form whatsoever.

Both parties exchange requests and responses that depend on private random numbers. A usual concept of proof is replaced by the probabilistic one, i.e. a proof is valid with certain probability, close to 1.

When authentication protocols based on the zero-knowledge concept are used, no security decline takes place with repeated use of the protocol.

Authentication problem is inexorably associated with data integrity and hash functions.

Hash Functions And Data Integrity

Hashing functions are extremely important for data integrity and authentication.

A binary sequence of arbitrary length is supplied at the hash function input, which produces a hash value at the output – a fixed-length sequence consisting of *n* bits. The principal concept of cryptographic hash functions implies that a hash function serves as a sort of an imprint or summary of a message.

Two types of hash functions exist:

- modification detection codes MDC (one-sided hash functions and hash functions in respect of which a search of various messages having the same value of hash function is a problem; such types of hash functions belong to unkeyed hash functions and they ensure verification of message integrity);
- message authentication codes (MAC) that ensure integrity verification of a message and its authentication (keyed hash functions).

Unkeyed hash functions must possess the following properties:

- input sequence, whose hash value is equal to a known value, is computationally impossible to determine (one-way property of a hash function);
- one more input sequence having the same value of hash function is computationally impossible to determine;
- two input sequences having the same value of hash function are impossible to determine.

Unkeyed hash functions are built on the basis of block ciphers or with the use of modular arithmetic and are usually used to ensure data integrity. The value of hash function for a certain message is protected in some way and is transmitted together with such message. The receiving party calculates the value of hash function for this given message and compares it with the received value of hash function. If two values are equal, it means that the message was not altered during transmission. Therefore, the large-size message integrity problem reduces to the integrity problem of *n*-bit value of hash function.

Keyed hash functions (MAC) are usually based on block cipher algorithms. There are a few suggested methods to create keyed hash functions from unkeyed hash functions, with a key used as a part of a message.

There are also hash functions constructed specially for implementation in specific computing machinery, as well as hash functions designed specifically for stream ciphers.

Message integrity means that a message was not altered by any third party during its creation, transmission or storage. Hash functions are used specifically for that purpose.

Message source authentication means that data has been received from an expected source. This task can be accomplished by using message authentication codes (MAC), digital signature schemes or by adding an authenticator to the message prior to its encryption. In this case, confidential keys are used for each subscriber in the communications network.

Transaction authentication is used to verify uniqueness and timeliness of data transmissions. In this case, random numbers, sequence numbers and time stamps are used.

Digital Signature

In modern information management systems, an electronic document turnover strongly necessitates a digital analog for a signature to documents. Such analog is represented by a digital signature that is a number dependent on a certain cryptographic key and contents of the undersigned message.

A digital signature is used for message authentication, data integrity and non-repudiation. However, certification of public keys in large communications networks is one of its most important applications.

Digital signature schemes are classified into two groups:

- schemes with appendix that require a source message to verify the digital signature (they are most common and are used in respect of messages of unrestricted length);
- schemes with message recovery that do not require a source message to verify the digital signature (it is recovered from the digital signature, and the schemes themselves are applicable to messages of short and fixed length and use public key encryption methods).

All digital signature schemes are divided into randomized and deterministic ones (i.e. those having a one-time digital signature and those having a reusable digital signature).

In addition, there exist digital signature schemes that use symmetrical keys and require availability of a third party, that is completely trusted by the other parties and serves to generate and verify signatures. There are also schemes that allow a transmitted message to be confidential to its sender.