International Research Journal of Engineering, IT & Scientific Research
Available online at https://sloap.org/journals/index.php/irjeis/
Vol. 2 No. 3, March 2016, pages: 16~32
ISSN: 2454-2261
https://sloap.org/journals/index.php/irjeis/article/view/484

Security: Hash Function-Authentications

Saroj Singh [a]

Article history:

Received: 27 December 2015
Accepted: 30 January 2016
Published: 31 March 2016

Keywords:

authentication;
firewall;
hash function;
security;
threats;

Abstract

As security or firewall administrator, we got basically the same concerns (as a plumber) the size of the pipe the contents of the pipe, making sure the correct traffic is in the correct pipes and keeping the pipes from splitting and leaking all over the places of course like plumbers. When the pipes do leak: we are the ones responsible for cleaning up the mess and we are the ones who come up smelling awful. Firewall is a device that is used to provide protection to a system from network-based security threats. The firewall uses service, behavior, user and direction control techniques.

Author correspondence:
Saroj Singh,
Dept: Computer Science & Engineering, Delhi Engineering College, Ladiyapur, Faridabad, India
Email address: sarojraj47@gmail.com, acme.singh10@gmail.com

1. Introduction

Authentication header provides integrity and authenticity of IP packets while encapsulating security payload provides confidentiality services. The firewall provides protection to a system from network-based security threats there are some techniques are:

Service Control	: Determines the type of services.
Direction Control	: Determines the direction in which the services are initiated.
User Control	: applied to internal and external users.
Behavior Control	: How services are used.

Digital signature [1] scheme is a mathematical method used for checking the authenticity, integrity, and non-repudiation of a document. Approaches proposed for digital signatures are direct and arbitrated digital signatures involves two communication parties (sender and receiver) while arbitrated digital signatures involve three communication parties (sender, receiver and arbiter).

Pretty good privacy (PGP) was created by Phil Zimmerman Authentication, confidentiality, compression; email compatibility, segmentation, and reassembly are the six operations of the PGP. PGP provides three keys symmetric, public and private key. These keys can be used by all communicating parties in an efficient manner.

[a] Dept: Computer Science & Engineering, Delhi Engineering College, Ladiyapur, Faridabad, India

2. Materials and Methods

This article is presented based on qualitative analysis. The data were obtained through observation and interviews. The observations were conducted in a non-participant manner and interviews were conducted in deep interview. The informants were determined purposively and snow ball. Data processing was done in three stages included data reduction, data presentation, and data verification/conclusion.

3. Results and Discussions

3.1 Protocols –Digital Signature Standard: Digital Signature Outline

Definition: Digital signature also known as digital signature scheme is a mathematical method for checking the authenticity of a message of a document. A valid digital signature gives the receiver a reason to believe that the message was created by a known sender and it has not been altered during the transmission from sender to receiver.

Requirements for a Digital Signature

1) The digital signature must be a bit pattern and this solely depends on the message being signed.
2) The signature must use some information unique to the sender to prevent both forgery and denial.
3) It should be easy to produce, recognize and verify the digital signature.
4) It should be practical to retain a copy of the digital signature in storage.

Properties of a Digital Signature

1) Authentication: Digital signature can be used to authenticate the source of message i.e. when a specific user is having the ownership of a secret key. A valid signature shows that the message was sent by that particular user. For example the head office of the bank sends an instruction to the branch office requesting some changes but the branch office is not sure whether the message is sent by the head office.
2) Integrity: The sender and receiver of the message should be in confidence that the message is not changed during its transmission. Even though encryption hides the contents of the messages but it is still possible to change the contents of the encrypted message thus if the message is digitally signed, any change in the message will make the signature invalid.
3) Non – Repudiation: If a person has signed the message, then later one cannot deny signing the message.

Approaches for a Digital Signature

A variety of approaches have been proposed for the digital signature function these basically fall into two categories:

1) Direct Digital Signature: It involves only the communication parties (source and destination). A digital signature is formed by encrypting the entire message with the sender’s private key or by encrypting hash code of the message with the sender’s private key and the receiver knows the public key of the source. In the case of depute, some third party must view the message and its signature. Confidentiality can be provided by encrypting three parts:
    a) Entire message
    b) Digital signature
    c) Receiver’s public key or shared secret key
2) Variants: If the signature is encapsulated on an encrypted message then the third party also needs access to the decryption key to read the original message. If the signature is an inner operation, then the recipient can store the plaintext message and its signature for later use in depute resolution.
3) A problem associated with direct digital signature: There is a need of trust between the sender and receiver since there is no independent verification process.

    The validity of the scheme depends upon the security of the sender’s private key. If a sender later wishes to deny sending a particular message then the sender can claim that the private key was lost or stolen and that someone else forged his or her signature.

    The private key might actually be stolen from X at time T. The opponent can send a message signed with X’s signature and stamped with a time before or equal to T.

4) Arbitrated Digital Signature: The problem faced by the digital signature is solved with the help of an arbiter. Every signed message sender X to receiver Y first goes into an orbiter A. the orbiter subjects the message and the signature to numerous tests to check its origin and contents. Then the message sends to Y with an indication that it has been verified by the orbiter. Every party should pass a great deal of trust with the orbiter mechanism.

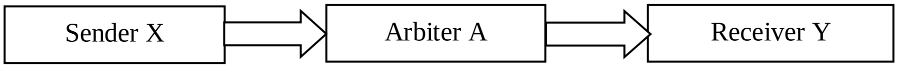

Figure 1: Arbitration Mechanism

Example: The orbited digital signature can be explained with the following steps:

a) Sender X and an orbiter ‘A’ share a secret key K_{xa}. The receiver Y and an orbiter ‘A’ share a secret key $K_{V.}$

b) Sender X constructs a message M and computes its Hash value H(M).

c) Sender X transmits the message and signature to A.

The signature consists of:

a) Identifier (ID_x) if X

b) Hash value encrypted using K_V

c) Arbiter A decrypts the signature and checks the hash value to validate the message. If correct, A transmitted the message to Y encrypted with $K_{AV.}$

The message consists of:

a) ID_x

b) Original message from X

c) Signature

d) Timestamp

Y can decrypt the message. Timestamp inform Y that the message is timely.

Problems associated:

a) Using an arbiter requires complete trust from both the sender and receiver, that the arbiters will not only timestamp and forwards the document as instructed. But also not alter the data in any way.

b) There is also the possibility that an arbiter may shoe preference towards one party or another. The arbiter can form a group with the sender to deny a signed message or with the receiver to forge the sender’s signature.

5) Difference:

Table 1
Difference

Direct Digital Signature	*Arbitrated Digital Signature*
Only two communication parties are involved (sender and receiver)	Three communication parties are involved (sender, receiver and arbiter)
The third party is involved only in the case of dispute.	The third party is involved during the entire process.

Attacks against Digital Signature Schemes

The attacks [2] can be classified in two ways:

1) Key only attacks: The attacker knows only the signer’s public key.

2) Message attacks: The attacker is able to examine the signature corresponding either to known or chosen messages. These are the three types:

   a) Known message attacks: An attacker is given a valid signature for a variety of messages known by the attacker but not chosen by the attacker.

   b) Nonadaptive chosen-message attack: An attacker obtains a valid signature from a chosen list of messages before attempting to break the signature scheme. This attack is non-adaptive because the messages are chosen before any signatures are seen.

c) Adaptive chosen-message attack: An attacker first learns signatures on arbitrary messages of the attacker's choice.

Digital Signature Standard

The national signature institute of standards and technology (NIST) [3] has published Federal Information Processing Standard FIPS PUB 186 known as the Digital Signature Standard (DSS) [4]. It was proposed in 1991 and was revised in 1993. It basically uses two techniques:

1) RSA Approach:

Definition: RSA stands [5] for Rivet, Shamir, and Adleman. The message to be signed is input to the hash function that produces a secure hash code of fixed length [6]. This hash code is encrypted using the sender's private key to form a signature. Then both the signature and message are transmitted. The receiver takes the message and produces a hash code and decrypts the signature using the sender's public key. If the calculated hash code matches the decrypted signature, the signature is accepted as valid.

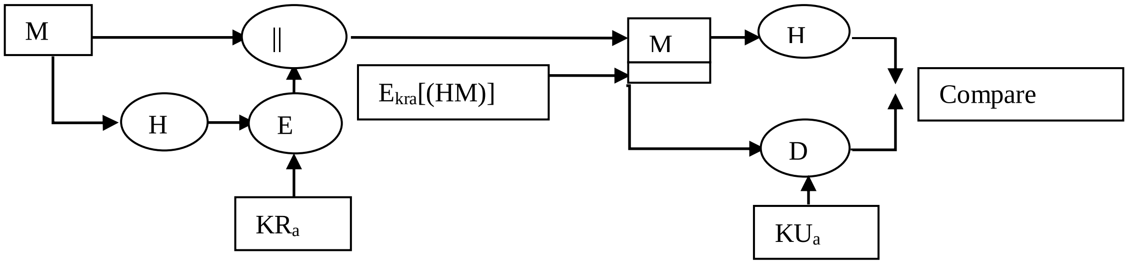

Figure 2: RSA Approach

a) How RSA operates: The operation of RSA can be explained in three steps.
b) Key Generation: There are two types of Key: Public and Private Key. The public key is one of that can be known to everyone and is used for encrypting messages. Private Key is used to decrypt the messages that are decrypted by the public key [7].
c) Steps for key generation:
d) Select two distinct and similar bit length prime numbers 'p' and 'q'.
e) Computer n=pq where n is the modulus of both the private and public key.
f) Compute $\varphi(n)=(p-1(q-1)$ where φ Euler's totient function.
g) Select an integer 'e' such that $1<e< \varphi(n)$ and $\gcd(e, \varphi(n))=1$where e is known as the public key exponent.
h) Compute $d=e^{-1} \bmod \varphi(n)$ such that $1<d< \varphi(n)$ and' is kept as the private key exponent.
i) Encryption: Suppose A is the sender and wants to send message M to receiver B. Then obtain the receiver B's public key (n, e) and represent the plaintext message M as a positive integer m Computer ciphertext by $c=m^e \bmod n$ and send it to the receiver B.
j) Decryption: Receiver B uses its own private key (n,d) to compute, i.e. $m=c^d \bmod n$ and extract the plaintext message from ciphertext message. For example of RSA: 1. Select two distinct prime numbers. For example, here we have considered p=31 and q=23
k) Now compute n=pq i.e. 31*23=713
l) Compute $\varphi(n)=(p-1)(q-1)=30*22=660$
m) Select any random integer. Here we have taken the value of e as 223

Now the main point comes in calculating the value of d. So,

660=223*2+214 or 214=660-2(223) ...(1)
223=214*1+9 or 9=223-214(1) …(2)
214=9*23+7 or 7=214-9(23) …(3)
9=7*1+2 or 2=9-7(1) …(4)
7=2*3+1 or 1=7-3(2) …(5)

Now from equation (5) we can get that

$$1=7-3(2)$$

Substituting equation (4) in above equation,

$$1=7-3(9-7)$$
$$1=7-3(9)+3(7)$$
$$1=4(7)-3(9)$$

Substituting equation (3) in above equation,

$$1=4(214-9(23))-3(9)$$
$$1=4(214)-36(23)-3(9)$$
$$1=4(214)-(23*9*4)-3(9)$$
$$1=4(214)-92(9)-3(9)$$
$$1=4(214)-95(9)$$

Substituting equation (2) in the above equation,

$$1=4(214)-95(223-214)$$
$$1=99(214)-95(223)$$

Substituting equation (1) in the above equation,

$$1=99(660-2(223))-95(223)$$
$$1=99(660)-293(223)$$

So, the inverse of 223 mod 660=-293=367
Thus private key=367 and public key=(223,713).
Encryption: Suppose sender wants to send message m=439

$$c=m^{e} \bmod n$$
$$c=439^{223} \bmod 713=284.$$

Sender sends the ciphertext 284 to the receiver.
Decryption: Receiver receives the ciphertext 284nfrom the sender.

$$M=c^{d} \bmod n$$

The receiver computer 284^{367} mod 713=439 and knows that the message is 439.
Use of RSA:

a) Software products like Apple and Microsoft.
b) Ethernet network cards.
c) Included in products for secure internet connection.

2) DSS Approach:

DSS Approach: DSS is a public key technique which cannot be used for encryption purpose. Here a random number ‘k’ and hashcode is provided as an input to the digital signature. The signature function depends upon two parts. First is the sender’s private key (KR_a) and second is the global public key (KU_g). The signature generated consists of two parts: ‘s’ and ‘r’. This all is performed on the sender’s side. Now on the receiver’s side, the hash code of the incoming message is generated. There is a verification function which depends upon the global key and the sender’s public key. It the signature is valid then the output of the verification function equals the signature component ‘r’.

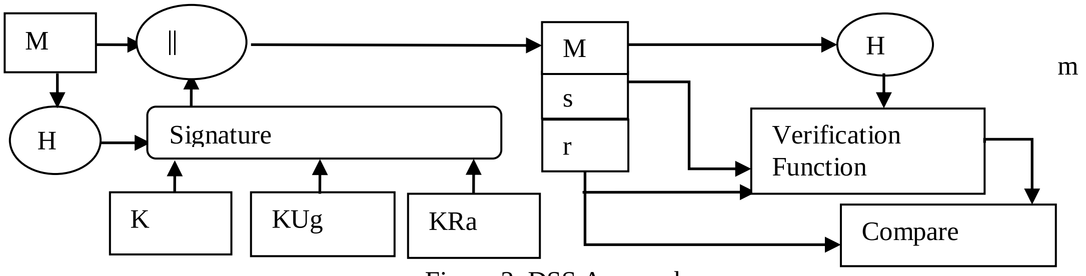

Figure 3: DSS Approach

Digital Signature Algorithm

The digital signature algorithm was proposed by the National Institute of Standards and Technology in August 1991. The digital signature algorithm can be explained in the following steps:

Key Generation: The key generation is performed in two parts. First is the choice of the parameters to be used in the algorithm and second is the computation of the public and private key for a single user.

1) The choice an appropriate hash function ‘H’. For example SHA-1 or SHA-2.
2) Choose the key length ‘L’ and ‘N’.
3) Choose L bit prime modulus ‘p’ of length between 512and 1024.
4) Choose N bit prime ‘q’: Prime q is a divisor of (p-1) where $2^{159} < q < 2160$.
5) Choose ‘g’: g should be of the form $h^{(p-1)/q}$ mod p where his an integer such that $h^{(p-1)/q}$ mod p>1.

Each user selects a private key and generates a private key corresponding to the number.

1) User’s private key: It is ‘2x’ which is random integer where 0<x<q
2) User’s public key: $y=g^x$ mod p.
3) User’s per-message secret number: It is ‘k’ which is a random number with 0<x<q.

Signing: The signing parts is done with the help of two components’ and ‘r’. ‘H’ is the hashing function and’ is the message. A random per message value ‘k’ is generated where 0<k<q.

1) $R=(g^k \text{ mod } p)\ q$
2) $S=[k^{-1}\ H(M) + xr] \text{ mod } q$
3) Signature=(r,s)

If in any, in any case, the value of ‘s’ or ‘r’ is 0 then start again with a random number ‘k’.

Verifying:

T verifying part is done on the receiver’s end.

1) Calculate $w=s^{-1}$ mod q
2) Calculate u1=H(M)w mod q
3) Calculate u2=rw mod q
4) Calculate $v=((g^{u1}\text{-}y^{u2}) \text{ mod } q$

The signature is valid if v=r.

3.2 Authentication Protocols

There are two general approaches. First is Mutual Authentication and second is one – way Authentication

Mutual Authentication

Definition: Manual authentication enables communicating parties to satisfy themselves mutually about each other’s identify and exchange session keys.

The two issues faced in mutual authentication are:

1) Confidentiality: The identification and session key information must be communicated in an encrypted form. Secret or public keys should be available.
2) Timeliness: There is a threat of message repays.

Example of Replay Attack:

1) Simple Replay: The opponents simply copy a message and repay it later.
2) Repletion that cannot be detected: the original message could have been suppressed and thus did not arrive at its destination. Thus only the repay message arrives.
3) Backward replay without modification: This is a repay back to the message sender. The sender cannot easily recognize the difference between a message sent and message received on the basis of content.

Approach to Cope with the Problems of Repay Attack:

1) Attach a sequence number to each message used in the authentication exchange. A new message is accepted only if its sequence number is in proper order.

Difficulty:

1) Requires each party to keep in the track of the last sequence number it has dealt with.

2) Timestamps: Party A accept a message only if the message contains a timestamp. The clocks among the various participants should be synchronized.
3) Challenge Response: the party that is expecting a fresh message from B first sends B a challenge and requires that the response received from B must contain the correct challenge value.
4) The timestamp approach should not be used for connection-oriented applications.
5) The challenge/response approach is not suitable for connectionless application.

Symmetric Encryption Approaches (Mutual Authentication):

1) A two-level hierarchy or symmetric encryption keys can be used be used to provide confidentiality for communication in a distributed environment.
2) Here trusted key distribution center (KDC) is used. Each party in the network share a secret key called master with KDC.
3) KDC is responsible for generating keys. These keys are used for a short time over the connection between two parties called session keys.

1) Need a protocol scheme:

a) Secret keys K_a and K_b are shared between A and KDC, and B and KDC.
   A->KDC: $ID_A \| ID_B \| N_1$
b) ‘A’ acquire a new session key:
   KDC-> A: $E_{Ka} [K_S \| ID_B \| N_1 \| E_{Kb} [K_S \| ID_A]]$
c) Now the message is decrypted only understood by B.
   A -> B: $E_{Kb} [K_s \| ID_A]$
d) Now B’s knowledge of K_S is reflected
   B-> A: $E_{KS} [N_2]$
e) Now B is assured of A’s Knowledge of K_S. It assumes that this is a fresh message because of the challenge N_2.
   A-> B: $E_{KS}[f(N_2)]$

Weakness: Suppose there is an opponent X that has been able to compromise the old session key. X can send bugs message to B that appear from B to A.

2) New approach followed: Including timestamp:

a) A-> KDC: $ID_A \| ID_B$
b) KDC-> A : $E_{Ka} [K_S \| ID_B \| T \| E_{Kb}[K_S \| ID_A \| T]]$
c) A-> B: $E_{Kb} [K_S \| ID_A \| T]$
d) B->A: EKS $[N_1]$
e) A->B: EKS$[f(N_1)]$

T is the timestamp that assures A and B that the session key has only just been generated. A and B can verify timeless by:

a) $|Clock – T|<\Delta t_1 + \Delta t_2$
b) Δt_1 estimated normal discrepancy between KDC’s clock and a local clock.
c) Δt2 expected network delay time.
d) The time stamp T is encrypted using secure master keys. Thus an opponent even with knowledge of an old session key cannot succeed.

Disadvantages: [Suppress-Replay Attack] there is a risk associated with the synchronization of the clock. Sender’s clock is ahead of the intended receiver’s clock. Thus the opponent can interrupt a message from the sender and replay it later. Thus causing unexpected results.
Solution: Parties should regularly check their clocks against the KDC’s clock.

3) NEUM 93 a Protocol:

a) A initiates the authentication exchange by generating a challenge N_a. N_a + Identifier of A is send to B in a plain text format. The challenge will be returned to A in an encrypted format that includes the session key.
A-> B: $ID_A||N_a$

b) B alerts the KDC that a session key is needed. Message includes:
1) Identifier ID_B.
2) Challenge N_B.
The challenge will be returned to B in an encrypted format that includes the session key.

$$B\text{-> } KDC\text{: } ID_B \mid\mid N_B \mid\mid E_{KB} [\ ID_A \mid\mid N_a \mid\mid T_B]$$

c) The KDC is then passed to A. It also sends A, a block that is encrypted with secret key shared between A and KDC. This block verifies that B has received A's initial message is a timely message.

$$KDC\text{->}A\text{:}E_{Ka}\ [ID_b||N_a||K_S||T_b]||E_{Kb}\ [ID_A||K_STb]||N_b$$

The block serves as a ticket.

d) A transmits the ticket to B with B's challenge that is encrypted with the session key. The ticket provides B with a secret key that is used to decrypt $E_{KS}[N_b]$ to recover the challenge.

$$A\text{-> } B\text{: } E_{KB}[ID_A||K_S||T_b][E_{KS}||N_b]$$

Advantages:

a) A and B can establish a secure and effective session with a secure session key.
b) A has a Key that can be used for subsequent authentication to B thus avoiding the need to B thus avoiding the need to contact the authentication [8] server repeatedly.

One Way Authentication

E-mail: it is the application for which the encryption is growing its popularity. The header of the e-mail message must be clear so that message can be handled by the store and forward e-mail protocol. There are some approaches followed:

1) Symmetric encryption [9] approach
Sender issues a request to the intended receiver and awaits a response that includes a session key and then sends the message.
This approach guarantees that only the intended receiver of a message will be able to read it. It provides a level of authentication that the sender is genuine.
This approach does not protect against reply attacks.
2) Public key encryption approach:
Here either the sender knows the receiver's public key (confidentiality) or the receiver knows the sender's public key(authentication).

A public key algorithm must be applied once or twice to the message which is big in size
Confidentiality: The message is encrypted with one-time secret key.
'A' encrypts the one time secret key with B's public key.
'B' recovers the one-time secret key by using the private key and then decrypts the message.
Advantages: It is efficient.
Authentication: Both the message and the signature is encrypted with the receiver's public key.
'A' encrypt the signature with A's private key and A's certificate is encrypted with the private key of the authentication server.
The receiver first uses the certificate to obtain the sender's public key and verifies that it is authenticated and then uses it to verify the message.

3.3 Prety Good Privacy

Definition: PGP [10] acronym of pretty good privacy was created by Phil Zimmermann in 1991 and is an encryption and decryption program that provides confidentiality and authentication service. These are basically used in email and file storage applications.

1) *Approaches:*

There are some approaches are followed:

a) Choose the best available cryptographic algorithm as the building block.

b) Combine these cryptographic algorithms into a general purpose application these integrated applications are independent of the operating system and the processor being used.
c) Create the package and document including the source code that is freely available on the internet.
d) Make an agreement with the company to provide a fully compatible and low-cost commercial version of PGP.

Applications

a) E-mail attachments
b) Digital Signature
c) File and folder security
d) Batch file transfer encryption
e) Protection of files and server stored on a network server

Versions

a) Unix PGP
b) MAC PGP 2.3
c) OS/2 PGP
d) Amiga PGP
e) MS-DOS PGP
f) Archimedes PGP
g) Atari PGP

Advantages of PGP

a) It is platform independent.
b) It is built on algorithms that are secure.
c) It provides wide range of applicability.
d) It is not controlled or developed by any government organization.

Notation of PGP

a) Ks: This is the session key used in the encryption scheme
b) KR_a: This is the private key of the user A and is used in the public key encryption scheme
c) KU_a This is the public key of the user A and is used in the public key encryption scheme.
d) EP: Public Key Encryption
e) DP: Public Key Decryption
f) EC: Conventional Encryption
g) DC: Conventional Decryption
h) H: Hash Function
i) ||: Concatenation

Operations of PGP

1) Authentication: Sender creates a message by using SHA-1 to generate 160-bit hash code of the message. This hash code generated is encrypted with the RSA using the sender's private key. The receiver uses the RSA with the sender's public key to decrypt and recover the hash code. The receiver generates the new hash code and compares with the decrypt hash code. If the two codes matches, the message is accepted as authentic.
2) Confidentiality: The encrypted messages are either transmitted or stored locally as files. The symmetric encryption algorithm CAST-128 and 64-bit cipher feedback mode are used. The sender generates a message and a random 128-bit number that is used as a session key. The message encrypted using CAST -128 with the session key. Here the session key encrypted with RSA using the recipient's public key and is attached to the message. The receiver uses RSA with its private key to decrypt and recover the session key the session key is used to decrypt the message. PGP offers an option Diffie Hellman instead of RSA for encryption.
3) Confidentiality and Authentication: A signature is generated for the plaintext message and is attached to the message. The plaintext message and the signature are encrypted using CAST-128 and the session key is encrypted using RSA.
4) Compression: PGP compresses the message after applying the signature but before encryption. The advantage of doing this is saving space for both email transmission and file storage. Firstly it is preferable to sign an

uncompressed message so that one needs to store only the uncompressed message before encryption. Secondly, strengthen security. Here the compression algorithm used is ZIP.

5) E-mail compatibility: Many of the email systems only permits the use of blocks consisting of the ASCII text. PGP provides the service of converting the raw 8-bit binary stream to ASCII characters. Here radix 64 conversions is used. Each group of three octets of binary data is mapped into four ASCII characters.
6) Segmentation and Reassembly: the email facilities are restricted to a maximum message length. PGP subdivides a message that too large into segments so that transmission can be sent via email. Segmentation is performed after all the other function including the radix 64-bit conversion is completed. The session key and the signature components appear only once at the beginning of the first component. At the receiving end, PGP must strip off all email headers and reassemble the entire message.

Open PGP

Definition: it is an an industry that uses a combination of asymmetric cryptography and symmetric cryptology. Cryptology is a secure method of communication that is performed in the presence of the third party it is the most widely used encryption technique nowadays and provides a high level of data security.

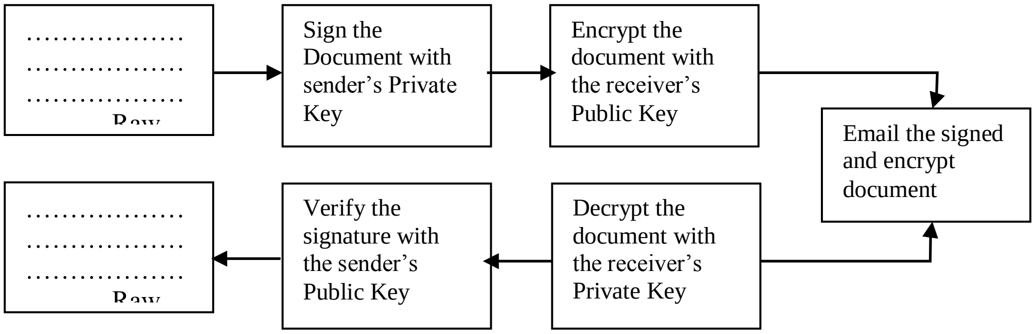
....
....
....
Raw
Sign the Document with sender’s Private Key
Encrypt the document with the receiver’s Public Key
Email the signed and encrypt document
Decrypt the document with the receiver’s Private Key
Verify the signature with the sender’s Public Key
....
....
....
Raw

Figure 4: Open PGP

Keys and Key Rings:

PGP makes use of three types of keys: Public, Private and symmetric keys. These keys are needed to be stored and organized in a symmetric way so that they can be used in an efficient way by all parties. Pair of data structures is used at each node. The first data structure is used to store the public/private key pair that is owned by the node and the second is used to store the public key of other users.

Session key generation: Each session key is associated with a single key and is used for the encryption and decrypt of the message. For example CAST – 128 and IDEA use 128-bit key and DBS uses 168-bit key.

Format of PGP message

The format can be basically classified into three distinct parts: session key component, signature, and message.

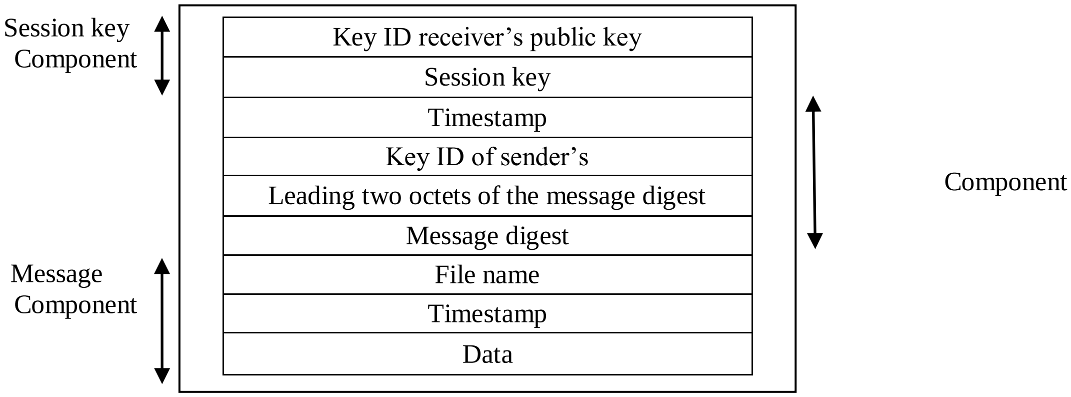

Figure 5: Format of PGP Message

Explanation of the entities:

1) Timestamp: it is the data and time when the pair was generated
2) Key ID: These are the least significant 64 bits of the public key.
3) User ID: this is the user's email address.

3.4 Mime

Definition: MIME is the multipurpose internet mail expression, an extension to the RFC 822 framework. It is basically an internet standard that extends the format of the email. It addresses the difficulties faced by the SMTP and overcomes the problems by providing appropriate solutions. For example one can use characters other than ASCII and attaching attachments other than text documents.

Limitations of SMTP: SMTP cannot be used for transfer of text data that includes national language characters and rejects mail messages over a certain size.

MIME Specifications: the MIME specification includes three entities:

1) Five new message header fields
2) Content Formats
3) Transfer Encodings

Header Files: there are five types:

1) MIME-Version: must have the parameter value 1.0 thus header appears as MIME-version 1.0
2) Content-Type: It describes the data that is present in the body in sufficient details. It appears as Content-Type "text/plain".
3) Content transfer encoding: It includes the type of transformation that has been used to represent the body of the message. It should be acceptable for mail transport.
4) Content ID: this is used to identify MIME entities and is used in multipart messages. In multi-part, each part is uniquely identified and is referred to as the content ID.
5) Content Description: It is the text description of the object is not readable. For example audio data.

MIME Content Types: These are the following:

1) Text: This can be unformatted text or enriched text. For example ASCII. It provides much flexibility.
2) Image: JPEG or GIF format.
3) Video: MPEG.
4) Audio: Its subtype is BASIC
5) Application: Its subtype is postscript and octet stream. For example Adobe postscript.

MIME Transfer Encodings: The main objective of mime transfer encoding is to provide reliable delivery of data across the largest network. It defines two methods of encoding data. It consists of six values on which encoding takes places:

1) 7-bit: the data is represented by short lines of ASCII characters. It basically provides some information about the nature of data.
2) 8-bit: this data is also represented by short lines but it is not necessary that these may be ASCII characters. It also provides some information about the nature of data.
3) Binary: Makes no use of short lines and ASCII characters.
4) Quoted-Printable: here encoding is done in such a way that both in ASCII format and human-readable form.
5) Base 66: these are ASCII printable formats and encoding is performed by mapping 6-bit blocks of input to 8-bit blocks of output.
6) X-Token: it is a nonstandard encoding and mostly vendor specific.

S/MIME Functionality: It provides four types of functions:

1) Enveloped Data: this consists of encrypted data and encrypted keys for one or more receiver. The encrypted data can be of any type or format.

2) Signed Data: firstly digital signatures is formed by taking the message digest of the content to be signed and then this digital signature is encrypted with the private key of the signer. Finally, the data content and digital signature are encoded by the base 64 encoding.
3) Clear Signed Data: here only the digital signature is encoded using base 64. The receiver that is not S/MIME capable can also view the message but cannot verify the signature
4) Signed and Enveloped Data: here nesting of signed data and encrypted data is done. Thus the encrypted data can be signed and signed data can be encrypted.

Cryptographic Algorithms Used in S/MIME: The algorithms used in S/MIME can be explained on the basis of two aspects: function and the requirement to complete that function.

1) Function: Creation of a message digest that is used in creating the digital signature.
2) Requirement: It must support SHA-1 and the receiver must support MD-5 for compatibility.
3) Function: Encryption of the message digest to form the digital signature.
4) Requirement: The sending agent must support DSS and RSA encryption and the receiving agent must support verification of the RSA signatures.
5) Function: Encryption of the session key and transmission of the message.
6) Requirement: Sending agent must support Diffie Hellman and RSA encryption which the receiving agent must support RSA decryption.
7) Function: Encryption of message for transmission with one-time session key.
8) Requirement: Sending agents and receiving agents should support triple DES for encryption-decryption purpose respectively.

Certificate processing for S/MIME: S/MIME is standard for public key encryption and uses public key certificates. The S/MIME cannot be used directly for any application one needs to obtain and install individual certificates for usage. There are some difficulties should be facing:

1) All email software does not handle S/MIME signatures.
2) It is not suitable for webmail based browsers.

User Agent Role: It has the following role:

3) Key Generation: the user must be capable of generating Diffie Hellman, DSS and RSA key pairs in a secure and efficient manner.
4) Registration: the public key of the user must be registered with a certificate authority in order to receive the X.509 certificate.
5) Storage and Retrieval of Certificates: The user requires an access to a local list of certificates for verifying the incoming signatures and encrypts the outgoing signatures. This list is either maintained by the user or local administrative on the behalf of the user.

Security Services: There are three security services provided:

6) Signed receipt:
7) Security labels
8) Security mailing lists

3.5 Mime Security with Prety Good Privacy

The MIME is specified in six linked RFC documents. These are:

1) RFC 2045
2) RFC 2046
3) RFC 2047
4) RFC 4288
5) RFC 4289
6) RFC 2049

In MIME security with PGP the multipart /signed and multipart encrypted parts are treated by the agents. This means that the data cannot be altered in any way.

Pretty Good Privacy Encrypted Data: Before performing encryption with PGP, the data should be written in MIME canonical format. The MIME encryption format consists of the body and headers.

The multipart encrypted must consist of two parts:

1) MIME body part MUST have a content type of "application PGP encrypted"
2) MUST contain the actual encrypted data. The data must be labeled with a content type of "application/octet-stream"

PGP Signed Data: The PGP signed data is denoted by "multipart/signed" content type with a "protocol parameter" which MUST have a value of "application/PGP signature".

The multipart signed consists of two parts:

1) The signed data should be present in canonical format. It should also include a set of appropriate content headers which describes data.
2) PGP digital signature labeled with a content type of "application /PGP signature".

When the PGP data is generated the following steps are involved:

3) The data should be signed must first be converted to its type/subtype specific canonical format.
4) Appropriate content transfer encoding must be applied.
5) MIME content headers are then encoded to the body each ending with a canonical sequence.
6) The digital signature must be calculated over:
    a) Data to be signed
    b) Set of content headers.
7) The signature generated MUST be detected from the signed data so that the process does not alter the signed data.

Upon receiving the signed data following steps must be applied:

1) Convert the line ending to the canonical sequence before verification of the signature.
2) Pass both the:
    a) Signed data and
    b) Content headers

Combined method: version 2.x of PGP allows both signing and encryption in one operation

1) Advantages:
2) Less time
3) Less overhead

Distribution of PGP public Keys

1) Content-Type: Application/PGP-keys
2) Required parameter: NONE
3) Original parameter: NONE

3.6 Mime Security with Open Pgp

The three content type for implementing security and privacy with PGP are:

1) "Application/PGP encrypted"
2) "Application/PGP signature"
3) "Application/PGP keys"

Multipart Encryption: Before applying the open PGP encryption, the data is written in MIME canonical format. Open PGP encryption data is denoted by "multipart/encryptor" content type and MUST have a "Protocol" parameter value of "application/PGP encrypted".

Two parts of Multipart Encrypted are:

1) Content type “application/PGP encrypted”. This consists of the control information. It must contain a “version: 1” field.
2) It MUST contain the encrypted data. This a labeled with the content type of “application /octet-stream”

Open PGP Signed Data: Open PGP signed data is denoted by “multipart/signed” content type with a “protocol” parameter MUST have a value of “application PGP” signature. It also must contain one hash symbol of the format “PGP-<hash identifier>”. The hash-identifier identifies the message integrity check algorithm. This algorithm is then used to generate the signature.

Two Part of Multipart Signed:

1) MUST contain signed data in MIME canonical format. A set of appropriate content headers must be included for describing the data.
2) MUST contain OPEN PGP digital signature which is labeled with a content type of application /PGP signature”.

When the PGP data is generated following steps are involved:

1) The data to be signed must first be converted to its “content-type” specific canonical format.
2) Appropriate content transfer encoding is applied.
3) MIME content headers are then added to the body. Whitespaces must be removed from signed material.
4) The digital signature must be calculated over both the data to be signed and a set of content headers.
5) The signature must be generated detached from the signed data.

Combined Method: Open PGP packet format describes a method for signing and encrypting data in single open PGP message

Advantages:

1) Increase computability with implementation non-MIME of open PGP
2) Less overhead.

3.8 Data Compression

Definition: Data compression [11] is a method of encoding data by using fewer bits than original encoding. It is useful since most of the data is redundant and thus reduces redundancy. Compression algorithm scans for repetitions I a program and develops a code to replace the repeated sequence. Decompression algorithm should be able to deduce the mapping between codes and sequence of data input. There are two types:

1) Lossless Compression: For example Spreadsheet and Text.
2) Lossy Less Compression: For example Image sound and video.

Others examples are:

1) ZIP: It is a freeware package that is written C language and runs on UNIX and other platforms.
2) LZ77(Lempel-Ziv-77): It is based upon the facts that words within a text are likely to get repeated.
3) LZZZ is a lossless data compression algorithm. It was published in a paper by Abraham Lampeland Jacob Ziv in 1977 and 1978. It is also known as LZ1 and LZ2.
4) LZ88: In LZ88 compression is obtained by replacing the repeated occurrence of data with references to a dictionary.

Lossless Compression:

Definition: lossless compression is a collection of data compression algorithms which allows exact original data to be reconstructed from compression data.

Uses:

1) ZIP file format
2) UNIX tool Zip
3) Used as a component with lossy data compression technologies

Algorithm Used:

1) Run Length Encoding: It provides good compression of data.

2) LZ78, LZW: It is used by ZIP and part of compression process of PNG and PPP.

Lossy Compression:
Definition: Lossy compression is the method of compressing data by discarding some part of it.
Uses:

1) Compression of multimedia data
2) Streaming media
3) Internet telephony

Types: there are two types:
Lossy transform codes:

1) A sample of the picture is taken
2) Pictures are there chopped into small segments.
3) It is then transformed into a new space.
4) It is finally quantized.

Lossy Predictive Codes:
In this, the previous or next data is used to predict the current sound/image sample.
Functions:
Cropped ->Rotate-> flipped-> converted to grayscale
Disadvantages:
Lossy compression suffers from generation loss i.e. repeated compression and decompression of files where:

1) The index is the pointer to the previous dictionary entry.
2) Characters are appended to the string which is represented by the dictionary or index.

ZIP: ZIP is a file format for data compression. It was created by PHIL KATZ in 1998.

Definition: ZIP file contains one or more files that have been compressed to a reduced file size.
Design:

1) The directory is placed at the end of a ZIP file. This identifies what files are in the ZIP and identifies where ZIP file is located.
2) Extra data.
3) Two copies of the directory structure which provides protection against attacks.

Central directory: it stores the list of the name of the entries stored in ZIP file and information about the entry.
Buffers: the data compression uses two buffers:

1) Sliding history buffer: It consists of the last N characters of the source code that has already been processed.
2) Look Ahead Buffer: It consists of the next L characters of the source code that will be processed.

This algorithm tries to matches two or more characters from the beginning of the look-ahead buffer to a string in the sliding history buffer. If no match is found the first character in the look-ahead buffer is output as a nine-bit character and is a shift to the sliding window. If a match is found, the algorithm continues to scan for the next large match. The matched string is then given as output which gives three entities that is an indicator, pointer, and length.
Drawbacks:

1) If the size of the window is large as compared to that of the window many best matches are removed.
2) The size of the window can be increased and that leads to an increase in the processing time of the algorithm.

4. Conclusion

Digital signature also known as digital signature scheme is a mathematical method for checking the authenticity of a message of a document. A valid digital signature gives the receiver a reason to believe that the message was created by a known sender and it has not been altered during the transmission from sender to receiver.

Conflict of interest statement and funding sources

The author(s) declared that (s)he/they have no competing interest. The study was financed by the authors.

Statement of authorship

The author(s) have a responsibility for the conception and design of the study. The author(s) have approved the final article.

Acknowledgments

Nobody can do anything without the grace of God. I first thank to "Lord Krishna" and "OMKAR" who blessed me to achieve my ambition by writing this research paper named "Security: Hash Function-Authentications". I also thank my husband, my children, my in-laws and my parents who supported me for accomplishing my goal. I would also like to thank my friends and my colleague who supported me in each step. I would like to thank my publishers who gave me the golden opportunity to write this research paper.

References

1) Chien, H. Y., Jan, J. K., & Tseng, Y. M. (2002). An efficient and practical solution to remote authentication: smart card. *Computers & Security*, *21*(4), 372-375.
2) Hu, Y. C., Johnson, D. B., & Perrig, A. (2003). SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks. *Ad hoc networks*, *1*(1), 175-192.
3) Khan, M. K., Zhang, J., & Wang, X. (2008). Chaotic hash-based fingerprint biometric remote user authentication scheme on mobile devices. *Chaos, Solitons & Fractals*, *35*(3), 519-524.
4) Lian, S., Sun, J., & Wang, Z. (2006). Secure hash function based on neural network. *Neurocomputing*, *69*(16-18), 2346-2350.
5) Liao, Y. P., & Wang, S. S. (2009). A secure dynamic ID based remote user authentication scheme for multi-server environment. *Computer Standards & Interfaces*, *31*(1), 24-29.
6) Tsai, J. L. (2008). Efficient multi-server authentication scheme based on one-way hash function without verification table. *Computers & Security*, *27*(3-4), 115-121.
7) Wang, Y. Y., Liu, J. Y., Xiao, F. X., & Dan, J. (2009). A more efficient and secure dynamic ID-based remote user authentication scheme. *Computer communications*, *32*(4), 583-585.
8) Wang, Y., Liao, X., Xiao, D., & Wong, K. W. (2008). One-way hash function construction based on 2D coupled map lattices. *Information Sciences*, *178*(5), 1391-1406.
9) Wegman, M. N., & Carter, J. L. (1981). New hash functions and their use in authentication and set equality. *Journal of computer and system sciences*, *22*(3), 265-279.
10) Xiao, D., Liao, X., & Deng, S. (2005). One-way Hash function construction based on the chaotic map with changeable-parameter. *Chaos, Solitons & Fractals*, *24*(1), 65-71.