ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Security System for DNS Using Cryptography

Dr. S.Thajoddin
Associate Professor in Mathematics,
Osmania College, Kurnool.

Arif Khan
Assistant Professor in Mathematics,
Osmania College, Kurnool.

ABSTRACT:

The mapping or binding of IP addresses to host names became a major problem in the rapidly growing Internet and the higher level binding effort went through different stages of development up to the currently used Domain Name System (DNS). The DNS Security is designed to provide security by combining the concept of both the Digital Signature and Asymmetric key (Public key) Cryptography. Here the Public key is send instead of Private key. The DNS security uses Message Digest Algorithm to compress the Message(text file) and PRNG(Pseudo Random Number Generator) Algorithm for generating Public and Private key. The message combines with the Private key to form a Signature using DSA Algorithm, which is send along with the Public key. The receiver uses the Public key and DSA Algorithm to form a Signature. If this Signature matches with the Signature of the message received, the message is Decrypted and read else discarded.

INTRODUCTION:

The Domain Name System(DNS) has become a critical operational part of the Internet Infrastructure, yet it has no strong security mechanisms to assure Data Integrity or Authentication. Extensions to the DNS are described that provide these services to security aware resolves are applications through the use of Cryptographic Digital Signatures. These Digital Signatures are included zones as resource records. The extensions also provide for the storage of Authenticated Public keys in the DNS. This storage of keys can support general Public key distribution services as well as DNS security. These stored keys enables security aware resolvers to learn the authenticating key of zones, in addition to those for which they are initially configured.

Keys associated with DNS names can be retrieved to support other protocols. In addition, the security extensions provide for the Authentication of DNS protocol transactions. The DNS Security is designed to provide security by combining the concept of both the Digital Signature and Asymmetric key (Public key) Cryptography. Here the Public key is send instead of Private key. The DNS security uses Message Digest Algorithm to compress the Message(text file) and PRNG(Pseudo Random Number Generator) Algorithm for generating Public and Private key. The message combines with the Private key to form a Signature using DSA Algorithm, which is send along with the Public key. The receiver uses the Public key and DSA Algorithm to form a Signature. If this Signature matches with the Signature of the message received, the message is Decrypted and read else discarded.

DNS Transactions:

DNS transactions occur continuously across the Internet. The two most common transactions are DNS zone transfers and DNS queries/responses. A DNS zone transfer occurs when the secondary server updates its copy of a zone for which it is authoritative. The secondary server makes use of information it has on the zone, namely the serial number, and checks to see if the primary server has a more recent version. If it does, the secondary server retrieves a new copy of the zone. A DNS query is answered by a DNS response. Resolvers use a finite list of name servers, usually not more than three, to determine where to send queries. If the first name server in the list is available to answer the query, than the others in the list are never consulted. If it is unavailable, each name server in the list is consulted until one is found that can return an answer to the query. The name server that receives a query from a client can act on behalf of the client to resolve the query.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Then the name server can query other name servers one at a time, with each server consulted being presumably closer to the answer. The name server that has the answer sends a response back to the original name server, which then can cache the response and send the answer back to the client. Once an answer is cached, a DNS server can use the cached information when responding to subsequent queries for the same DNS information. Caching makes the DNS more efficient, especially when under heavy load. This efficiency gain has its tradeoffs; the most notable is in security.

DNSSEC:

In 1994, the IETF formed a working group to provide security extensions to the DNS protocol in response to the security issues surrounding the DNS. These extensions are commonly referred to as DNSSEC extensions. These security enhancements to the protocol are designed to be interoperable with non-security aware implementations of DNS. The IETF achieved this by using the RR construct in the DNS that was purposely designed to be extensible. The WG defined a new set of RRs to hold the security information that provides strong security to DNS zones wishing to implement DNSSEC. These new RR types are used in conjunction with existing types of RRs.

This allows answers to queries for DNS security information belonging to a zone that is protected by DNSSEC to be supported through non-security aware DNS servers. In order to gain widespread acceptance, the IETF DNSSEC WG acknowledged that DNSSEC must provide backwards compatibly and must have the ability to co-exist with non-secure DNS implementations. This allows for sites to migrate to DNSSEC when ready and allows less complexity when upgrading. This also means that client side software that are not DNSSEC aware can still correctly process RRSets received from a DNSSEC server [CHAR]. In March of 1997, the Internet Architecture Board (IAB) met to discuss the development of an Internet security architecture.

This meeting identified existing security mechanisms and those that are under development, but have not yet become standards, that can play a part in the security architecture. They also identified areas in which adequate protection using existing security tools could not be achieved. The results of this workshop include the identification of core security requirements for the Internet security architecture. Among those security protocols identified as core is DNSSEC. The protection that DNSSEC provides against injection of false cache information is crucial to the core security requirements of the Internet [RFC 2316].

DNS Transaction and Request Authentication:

DNS transaction and request authentication provides the ability to authenticate DNS requests and DNS message headers. This guarantees that the answer is in response to the original query and that the response came from the server for which the query was intended. Providing the assurance for both is done in one step. Part of the information, returned in a response to a query from a security aware server, is a signature. This signature is produced from the concatenation of the query and the response. This allows a security aware resolver to perform any necessary verification concerning the transaction. Another use of transaction and request authentication is for DNS Dynamic Updates. Without DNSSEC, DNS Dynamic Update does not provide a mechanism that prohibits any system with access to a DNS authoritative server from updating zone information. In order to provide security for such modifications, Secure DNS Dynamic Update incorporates DNSSEC to provide strong authentication for systems allowed to dynamically manipulate DNS zone information on the primary server [RFC 2137].

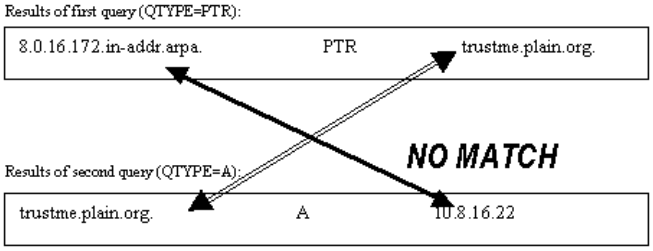

Figure . Example of a DNS cross check that fails

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Signature Generation

1. Select a random number k to be used only once, that is, for every new signature generation of a message, a new k is selected, such that 1 < k < n-1.
2. Generate (r, s) component of signature such that a. k.G = (x, y) r = x modulo n if r = 0 then repeat 2 again b. Calculate hash of message (M) whose signature is to be generated, i.e., e = h (M). c. s = d(r*k – e)-1modulo n // (modified)

Signature Verification

1. Calculate u1 = e*r-1 modulo n // (modified) 2. Calculate u2 = (r*s)-1 modulo n // (modified) 3. Calculate T = u1.G + u2.Q = (x1, y1), where '.' Is point multiplication and '+' is point addition and can be calculated using elliptic curve arithmetic.
4. Calculate v = x1 modulo n
5. If v = r, signature is valid. The above proposed algorithm is a variant of the algorithms as described in , providing less complexity in signing.

Security Need As

originally designed, DNS has no means of determining whether the domain name data comes from the authorized domain owner or it has been forged. This weakness in security leaves the system to be vulnerable to a number of attacks, like DNS cache poisoning, DNS spoofing etc. Due to weak authentication between DNS servers exchanging updates an attacker may predict a DNS message ID and manage to reply before the legitimate DNS server, thus inserting a malicious record into DNS database. The exploit forces a compromised DNS server to send a request to an attacker's DNS server, which will supply the wrong host to IP mapping. DNS Security Extensions (DNSSEC) is a set of IETF (Internet Engineering Task Force) standards which have been created to address the vulnerabilities in the DNS and to protect from online threats. The main purpose of DNSSEC is to basically increase the Internet security as a whole by addressing and resolving DNS security weaknesses.

Essentially, DNSSEC adds authentication feature to DNS that make the system more secure DNSSEC core elements were specified in following three IETF Requests for Comments which have been published in March 2005: • RFC 4033 - DNS Security Introduction and Requirements • RFC 4034 - Resource Records for the DNS Security Extensions • RFC 4035 - Protocol Modifications for the DNS Security Extensions Existing proposals for securing DNS are mainly based on public-key cryptography. The public key algorithms used for authentication in DNSSEC are MD5/RSA (Rivest Shamir Adleman Algorithm) and DSA (Digital Signature Algorithm). Digital signatures generated with public key algorithms have the advantage that anyone having the public key can verify them. The Idea behind it is that every node in Domain Name Space has a Public Key and each message from DNS Servers is signed using Private Key. Since DNS is Public, Authenticated DNS root Public Keys are known to all, which are used to generate Certificates/Signatures to combine the identity information of Top Level Domain. So, in Domain Name Space each parent signs the Public Keys of all its Children in the DNS tree.

Securing DNS with ECC

With the technology growing faster everyone accesses internet through mobile phones whether it is used to check EMails or visiting any secure sites, ECC (Elliptic Curve Cryptography) can be implemented. ECC provides same level of Security as RSA with benefits of small key sizes, faster computation, and memory and energy savings.

- Small Key Size and Faster Computation: The security level of 160-bit ECC and 1024- bit RSA is same. RSA operations are based on modular exponentiations of large integers and security is based on factoring these large integers. On the other hand, ECC operations are based on groups of points over elliptic curves and security is based on discrete logarithm problem (ECDLP). This allows ECC to have the same level of security with smaller key sizes and higher computational efficiency.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

• Memory and Energy savings: ECC requires less power for its functioning so it is more suitable for low power applications such as handheld and mobile devices. On small processors, multiple precision multiplication of large integers (done in RSA) not only involves arithmetic operations, but also a significant amount of data transport to and from memory due to limited registers space. While in ECC, the scalar multiplications involve additions with no intermediate results to be stored, thereby requiring less use of registers. So, ECC provides less memory space and also energy required to perform additions is much less than performing multiplications, done in RSA.

CONCLUSION:

The DNS as an Internet standard to solve the issues of scalability surrounding the hosts.txt file. Since then, the widespread use of the DNS and its ability to resolve host names into IP addresses for both users and applications alike in a timely and fairly reliable manner, makes it a critical component of the Internet. The distributed management of the DNS and support for redundancy of DNS zones across multiple servers promotes its robust characteristics. However, the original DNS protocol specifications did not include security. Without security, the DNS is vulnerable to attacks stemming from cache poisoning techniques, client flooding, dynamic update vulnerabilities, information leakage, and compromise of a DNS server's authoritative files. In order to add security to the DNS to address these threats, the IETF added security extensions to the DNS, collectively known as DNSSEC. DNSSEC provides authentication and integrity to the DNS. With the exception of information leakage, these extensions address the majority of problems that make such attacks possible. Cache poisoning and client flooding attacks are mitigated with the addition of data origin authentication for RRSets as signatures are computed on the RRSets to provide proof of authenticity. Dynamic update vulnerabilities are mitigated with the addition of transaction and request authentication, providing the necessary assurance to DNS servers that the update is authentic. Even the threat from compromise of the DNS server's authoritative files is almost eliminated as the SIG RR are created using a zone's private key that is kept off-line as to assure key's integrity which in turn protects the zone file from tampering. Keeping a copy of the zone's master file off-line when the SIGs are generated takes that assurance one step further. DNSSEC can not provide protection against threats from information leakage. This is more of an issue of controlling access, which is beyond the scope of coverage for DNSSEC. Adequate protection against information leakage is already provided through such things as split DNS configuration. DNSSEC demonstrates some promising capability to protect the Internet infrastructure from DNS based attacks. DNSSEC has some fairly complicated issues surrounding its development, configuration, and management. Although the discussion of these issues is beyond the scope of this survey, they are documented in RFC 2535 and RFC 2541 and give some interesting insight into the inner design and functions of DNSSEC. In addition to keep the scope of this paper down, many topics such as secure zone transfer have been omitted but are part of the specifications in RFC 2535. The first official release of a DNSSEC implementation is available in BIND version 8.1.2.

REFERENCES:

1. Albitz, P. and Liu, C., (1997) 'DNS and Bind', 2nd Ed., Sebastopol, CA, O'Reilly & Associates, pp.1-9.

2. Herbert Schildt, Edition (2003) 'The Complete Reference JAVA 2' Tata McGraw Hill Publications

3. IETF DNSSEC WG, (1994) 'DNS Security (dnssec) Charter', IETF.

4. Michael Foley and Mark McCulley, Edition(2002) 'JFC Unleashed' Prentice-Hall India.

5. Mockapetris, P., (1987) 'Domain Names -Concepts and Facilities'.