Secure Online DNS Dynamic Updates: Architecture and Implementation

Xunhua Wang, James Madison University, USA
David Rine, George Mason University, USA

ABSTRACT

Domain Name System (DNS) is the Internet infrastructure for mapping human-friendly domain names into IP addresses. To provide data-origin authentication for DNS, the DNS Security Extension (DNSSEC) standard was developed. In this article, we point out two drawbacks of DNSSEC in its handling of DNS dynamic updates: 1) creating a single point of attack with the on-line storage of a zone security key, and 2) violating the role separation principle by mixing up the roles of zone security managers and DNS name server administrators. To address these issues, we propose an alternative secure DNS architecture based on threshold cryptography. To demonstrate the feasibility of the proposed architecture, we developed a toolkit and built a proof-of-concept prototype. Our running results show that the performance of our architecture ranges from one to four times of DNSSEC’s performance. Thus, through small performance overhead, our architecture could achieve very high level of security.

Keywords: data security; digital signature; security management

INTRODUCTION

The Domain Name System (DNS) is a distributed database used in the Internet to map easily memorizable host names to their respective IP addresses (Mockapetris, 1987a, 1987b; Mockapetris & Dunlap, 1986, 1988, 1995). The DNS name space is organized into a hierarchy. Top-level domains include .com, .edu, .org, .biz, .info, .mil, .gov, .net, two-letter country codes like .ae and .jo (Postel, 1994). Second-level domain names typically designate individual institutions. For instance, Google Inc. is assigned a second-level domain name “google” under the top-level domain .com. In the domain name hierarchy, the subspace that is under a single administrative control is called a zone. In each zone, several predefined resources can be associated with a given domain name. Two example domain name resources are IP address and mail exchange server. The association of a domain name with a resource is called a resource record (RR). The most important RR of a domain name is the “type A” RR, which contains the host IP address of the domain name. All the RRs within a zone are stored in a master file to be published by
the primary name server of that zone. Each zone also supports zero or more secondary name servers, which obtain RRs from the primary server. Secondary servers act as backup of the primary name server and can also reduce the workload of the primary server; they send appropriate RRs to clients in response to queries but are not involved in the maintenance of the master file.

Unfortunately, the DNS, a critical infrastructure component of the Internet, was designed without security considerations. In particular, the original DNS architecture provides no way for a client to authenticate a received RR. This loophole enables many security attacks (Bellovin, 1995; Schuba, 1993; Vixie, 1995). For example, an attacker in the middle can modify a DNS response to include a fake RR. By providing an incorrect IP address for the requested domain name (for instance, www.ebay.com), a malicious third party could cause the loss of business to the domain name owner (eBay Inc. in the example).

In response to the above concerns, the DNS Security Extension (DNSSEC) was developed by the Internet Engineering Task Force (IETF) (Arends, Austein, Larson, Massey, & Rose, 2005a, 2005b, 2005c). Throughout this article we use the terms IETF DNSSEC and DNSSEC interchangeably. The DNSSEC provides RR authentication by the use of digital signatures (Diffie & Hellman, 1976; Rivest, Shamir, & Adleman, 1978). With DNSSEC, each zone is equipped with a public/private key pair. Resource records of a zone with the same name are organized into RR sets (RRset) and the zone private key is used to digitally sign all the RRsets in that zone. For each RRset, its digital signature is stored in a newly defined resource record called RRSIG RR. The response to a DNS query comprises the requested RRsets and the corresponding RRSIG RR. The zone public key is disseminated through DNSKEY RR, a new RR defined by DNSSEC, and DNS clients use this key to verify RRSIG RRs. (Obviously, a zone’s DNSKEY RR should be also authenticated and this is accomplished by a corresponding RRSIG RR by its parent zone.)

A major security goal of the DNSSEC is the role separation (Sandhu, Bhamidipati, & Munawer, 1999; Sandhu, Coyne, Feinstein, & Youman, 1996): It differentiates the roles of the DNS zone manager from the DNS name server administrator. In DNSSEC, it is the DNS zone manager, not the DNS name server administrator, who is responsible for the security of a zone. A DNS server is just a place to publish the digitally signed DNS zone data. With this separation, the compromise of secondary servers in a zone or even that of the primary server itself (assuming the zone private key is kept off line) will not necessarily affect the degree of assurance that a DNS client receives (Arends et al., 2005a). Role separation is consistent with the restriction idea in the design of secure systems (Saltzer & Schroeder, 1975) and meets the principle of least privilege and the principle of separation of privilege (Saltzer & Schroeder, 1975). The principle of least privilege states that a subject should be given only those privileges that it needs to complete its task (Bishop, 2002, p. 343). The principle of separation of privilege states that a system should not grant permission based on a single condition (Bishop, 2002, p. 347). This role separation design in DNSSEC also introduces management flexibility: it enables multiple zones to share a single DNS server to publish their zone data while each individual zone manager enforces the security of its respective zone independently. When zone data needs
Ranking Web Services using Web Service Popularity Score
[www.igi-global.com/article/ranking-web-services-using-web-service-popularity-score/115936?camid=4v1a](www.igi-global.com/article/ranking-web-services-using-web-service-popularity-score/115936?camid=4v1a)

An Efficient and Accurate Discovery of Frequent Patterns Using Improved WARM to Handle Large Web Log Data