Asian Journal of Engineering and Applied Technology (AJEAT)
Vol.2.No.1 2014pp 9-11
available at: www.goniv.com
Paper Received :05-03-2014
Paper Published:28-03-2014
Paper Reviewed by: 1. John Arhter 2. Hendry Goyal
Editor : Prof. P.Muthukumar

PREVENTING DNS SPOOFING ATTACK IN LOCAL AREA NETWORK USING DNS PROXY

KalaimaniBalu, Mr. M. Vivekandan
Department of Information Technology
SRM university, kattankulathur, Chennai, Tamil Nadu, India
Kalaimani.balu@yahoo.com Indiavivekandan@ktr.srmuniv.ac.in

ABSTRACT

DNS is one of essential part in internet communication. DNS resolves IP address for particular hostname.If DNS fails to do this the whole communication will stuck. But other problem is it has to resolve correct IP address, if it is a false IP address, further traffic will be redirected to fake domain and it leads to be vulnerable for other web based attacks. Making DNS to provide false response is called DNS poisoning, unfortunately DNS is highly vulnerable to these poisoning attacks. There are two major categories of DNS poisoning 1.OFF PATH attack(Cache poisoning), 2. ON PATH attack(Spoofing by MitM)Number of solutions have been proposed to prevent OFF PATH attack but solutions for ON PATH attack is limited. In this paper a proposed solution describes that A DNS proxy is used to prevent DNS spoofing attack in local area network. This proxy involves an acknowledgement system for dns queries. For each resolution of DNS query, DNS client has to acknowledge proxy with the IP address in the answer field. By this ACK proxy can verify whether IP address is spoofed or not. If it is so, proxy alerts client with re-ACK and original IP address. This method doesn't need any changes in current DNS server, so it is easy to implement when comparing other solution.

Index terms **–** DNS, DNS spoofing, MitM attack

1. INDRODUCTION

DNS is a layer wise distributed database that resolves IP address for requested domain name. Without DNS we have to memorize all IP address of webservers in order to view websites which is not possible at all. DNS has systematic query forwarding mechanism and components to do that.There are two major categories in DNS servers, 1.Recursive, 2.Iterative. Subcategories of DNS are Root server, TLD server, ANS, and RNS. DNS client which is called Stub-Resolver that generates the DNSqueries for the requested Domain name. This query reaches the local DNS which is also called RNS(Recursive Name Server). RNS looks through its own cache for IP address of requested domain name. If it is found in cache it replies the DNS client with the answer.If not found, RNS forwards the query to Root server. Root server replies RNS with the IP of corresponding TLD(Top[Level Domain) server of requested domain. For example .com, .net, .org and etc. RNS makes the same query towards TLD server and TLD server replies the RNS with IP address of corresponding ANS server of requested domain. The RNS sends the same query to ANS, and ANS replies the RNS with IP address of the host of requested domain name. RNS stores the answer in local cache for a TTL to reply future queries for same domain. UDP protocol is used to make this queries and 16 bit ID in this UDP packet to authenticate the response. If the ID is compromised thenattacker can send a forged response, which can poison the RNS cache, the all the clients may redirected to false domain, source port randomization is one of reliable solution for this OFF path attack and other solutions were also proposed for this attack.

There is another type attack which is called ON PATH which is done by Man in the Middle attack to spoofconnection between DNS server and DNS client or another DNS server. With some tools like Cain&Abel and Etter cap attacker can perform this attack and bypass all the DNS queries through his computer from which he can monitor the traffic and forge the DNS replies towards the DNS client. This attack mostly happens on LAN and focus on specific target computer. DNSSEC is a good solution for this problem but it hard to implement because it needs additional field in DNS packet and it should be patched both client and server side. Only 0.2%

system are patched with DNSSEC, most of them are Root servers and TLD servers.Here the proposed method is to resolve this problem by using a DNS proxywith no changes in current system.

2. ANTIDOTES FOR MITM ON DNS

SIG(0) provides a protection mechanism for DNS queries/answers and request whichis not provided by the usual SIG, KEY, and NXT RRs. But there is no mechanism to ensure over all integrity of a response packet and no protection for message headers. It provides transaction authentication which means that a client can be sure that it is getting the responses from the DNS server which it queried and that the received responses are for queries which are sent by Server.A SIG(0) are attachedin at end of resource record of the DNS response and it's corresponding query.

TSIG is mostly used between servers to authenticate each end points of the connection to make or respond a DNS transaction. It uses shared secret keys and one way hash function to secure the connection. Though queries to server may come anonymously, updates to servershould be authenticated because they make changes to Resource Records(RR). TSIG makes a final record which contains a timestamp and hash of the response and also contains identification of the secret key which was made to sign the queries. The reply to TSIG update is also signed using a TSIG record. Failure of transactions are left not signed to prevent that an attacker from observinginformation about the keys.

DNSSECprevents DNS client from using spoofed or manipulatedresponses. All responses from DNSSEC enableddomains are digitally signed. By verifying those signatures, the DNS clientscan check if the information at DNS resource records are information updated by the domain owner and distributed on original ANS. DNSSEC does not care about confidentiality of records, Responses from DNSSEC enabled domains are authenticated but not encrypted and it does not preventsDoS attack on servers. The DNSKEY record is authenticated through root zone which considered astrusted third party. All domain owners generate their own keys and update them by using their control panel at their domain-name registrar, which loads the key through secure DNS zone operator who signs and publishes them in DNS.

Additional fields of DNSSEC

RRSIG – signature of Resource record
DNSKEY – key to verify RRSIG
DS – list of delegated zone
NSEC – list of existing record types
NSEC3 - hashes of existing record types
NSEC3PARAM – verifies non exiting record types

WSEC-DNS (Wildcard Security) DNS, a strategy involving changes of resolvers and of DNS zone records (describing the mappings for a domain), that may deliver significant development in the defence against DNS poisoning by spoofing adversaries. WSEC DNS also avoids poisoning of resolvers minor in hierarchy, since requests are randomised, in contrast to formerly presented mechanisms. However, a necessity for modifications at both ends to the DNS transaction is aimportantproblem for deployment, similar to that of cryptographic protection against MitM (like DNSSEC), but in contrast to cryptographic protections it offers security only against off-path spoofs (not on-path). We classify two flaws in WSEC DNS and suggest countermeasures. WSEC DNS uses wildcard resource records and arbitraryrequests to dramatically increase the arbitrary space of encounters. Consider a FQDN *x* in the domain; to maintain WSEC DNS for this zone, the administrator have to include two wildcard RRs. The first Resource Record is for the wildcard domain name *._TEST_._WSECDNS_.*x*, and yields a TXTrecord containing the fixed string '|wsecdns=enabled|'; this permitsthe DNS clients to identify that the domain are protected by WSEC DNS. The second Resource Record is for the wildcard domain name *._WSECDNS_.*x*, and yieldsthe RR requested by the resolver.

3. PREVIOUS SECURE PROXY ARCHITECTURE

DNS system is not designed with respect to security, as DNS is not able to filter the forged packet. Because of this flaw attacker can poison the DNS cache. This secure proxy is deployed on both RNS and ANS. All DNS trafficsare normally routed between RNS and ANS like normal proxy. But it provides a secure environment to the transaction and also helps to detect the forged DNS server response. DNS proxy are categorised into two 1.Local proxy and 2.Remote proxy. Local proxy is deployed on the local DNS and the remote proxy is implemented on the ANS. All the DNS traffic passed through RNS and ANS with an exception that the DNS request to RNS then resply of RNS to client.

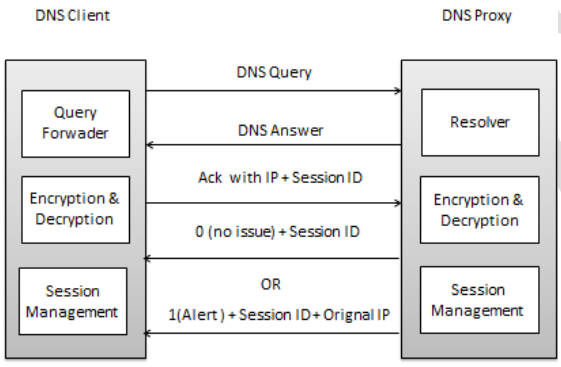
4. FORGERY DETECTION MECHANISMS

The 'forgery detection' mechanisms follow two methods: the combined approach and practices from machine learning. According to the combined approach the legitimacy of a DNS reply is validated by issuingthe DNS queries across hosts in the system, or by referring a set of reliable peers, and then, e.g., taking the most of answer. The common failings of the combined approach are most remarkably the performance penalty, i.e., additional handling and communication delaythat they present

to all DNS queries, even when the DNS is *not* under attack, and the importantsubstructure that is required for implementation. A recent system, by Antonakakisinvolvesmethods from machine learning to detectmalicious IP addresses. Precisely, designed a unified poisoning detection method is calledAnax, which is based on the reflexion that DNS records direct clients to a known set of NS records,while forgedrecords redirect users to false IP addresses,outside of the target's address space.However, implementation requires trust in one central entitythat should be referred to establish authenticity of theDNS replies. In addition, this method also results delays and may have false positives, e.g., if an ANSserver was moved to a new IP address for load circulation.

All solutions which are explained above for the ON PATH attack need changes current DNS system which makes them hard to be implemented. So here the proposed solution provides new, simple and efficient method to prevent this attack and can be easily implemented without any changes current DNS data fields.

DNS Proxy Architecture



In proposed method there is DNS proxy between the DNS server and DNS client. Actually there is no acknowledgement system in current DNS communication. DNS proxy has Resolver for DNS query, Encryption system for ACK packets and Session management system for synchronization with clients. Session is dedicated connection with a particular client to share the secret key to encryption and decryptions and to notify each other when any changes or updates happen on both side. After a successful establishment of a session with a client, the client sends all the queries to the proxy, and it resolves all the queries as normal and replies the client. The client has to send ACK with session ID and IP address from the answer. Data in ACK packet must be encrypted with the secret key in order to prevent sniffing by attacker. Proxy can decrypt this ACK and verify whether the IP address is spoofed in the answer by cross checking with resolved IP address. If so, then proxy sent alert message to client with session ID and original IP address, if not so proxy sends a no issue message with the session ID. Alert and no issue messages also encrypted to avoid sniffing. If client receives no issue message it proceeds with answer, if it is alert then client replace the IP address in the answer with the IP address in alert message and then proceeds further. In case the DNS answer is forged our proposed system can fix that issue and make it to original state. All above transaction are done by using UDP packets in order to reduce the time delay.

5. CONCLUSION

Even though number of solution has been proposed against DNS spoofing attacks, none of them deployed widely and each solution has their own pros and cones. Most of their drawback is they need additional field in DNS packet and all server must be able to identify those fields to make chain of trust.But this new system simple and capable enough to prevent spoofing attack with no need of additional fields in DNS packet

REFERENCES

[1]. TalhaNaqash, Faisal Bin Ubaid, AbubakarIshfaq, Fazal-e-Hadi, IEEE – Protecting DNS from cache poisoning attack by using secure proxy.

[2]. Amir Herzberg and Haya Shulman,IEEE – Antidotes for DNS poisoning by off – path Adversaries, 2012 Seventh International Conference on Availability, Reliability and Security

[3]. IEEE - Formal Analysis of the Kaminsky DNS Cache-Poisoning Attack Using Probabilistic Model Checking

[4]. Wikkipedia – DNSSEC, TSIG, SIG(0),WSEC-DNS

[5]. D. J. Bernstein. djb-dns, http://cr.yp.to/djbdns.html

[6]. Secure Domain Name System (DNS) Deployment Guide, NIST – United States of America.+/