Security Issues Related with DNS Dynamic Updates for Mobile Nodes: A Survey

Sadaf Yasmin
Center of Research in Networks and Telecom (CoReNeT), M. A. Jinnah University, Islamabad
sykahoot@yahoo.com

Muhammad Yousaf
Center for Advanced Studies in Engineering (CASE), Islamabad
myousaf@ymail.com

Amir Qayyum
Center of Research in Networks and Telecom (CoReNeT), M. A. Jinnah University, Islamabad
aqayyum@ieee.org

ABSTRACT

In today's heterogeneous wireless environment mobile nodes can move across different networks. This movement can cause the IP address of the mobile node to change, making it unreachable for other nodes in the network. In order to be available for other nodes, mobile node needs some location management mechanism. DNS dynamic updates have been proposed to be used as location manager for many transport layer mobility management protocols. Although secure dynamic DNS updates are used for location management, we have found that these secure updates are also susceptible to certain security issues. In this paper, we have highlighted many of these issues and discussed solutions that address these security issues. It has also been found that many of these attacks are interdependent on each other.

Keyword

DNS, Dynamic Updates, DNS Security Issues, Location Tracking

1. INTRODUCTION

In a mobile wireless environment, nodes have the flexibility to move from one network to another. When a mobile node moves to another network, it can no longer use the old IP address. In order to be able to receive traffic from Internet it has to acquire the new IP address in the new network.

Location management is necessary for a mobile node in order to be able to provide services to other nodes in spite of the change in its IP address. In Mobile IP, node updates its binding with the home agent in order for the traffic to be routed to its new IP address. Protocols in which no entity is used in the home network to support mobile node's movement DNS dynamic updates [13] can be used to update the current location of mobile devices. Other nodes will be able to locate the new IP address of the node by resolving domain name to IP address mappings. This enables the mobile node to change its IP address while keeping the same domain name within a domain. In case of static hosts, providing services to other nodes, location updates are also necessary if the host gets its IP address through DHCP server. This is done to allow other nodes to be able to connect with the host on the currently allocated IP address.

DNS was initially deployed to resolve domain name to IP address mappings. As it is not possible for human beings to remember IP addresses, therefore domain names are used to identify different nodes on the Internet. When a node wants to connect to another node, it first sends a query to the name server to get the corresponding IP address. Instead of manually updating DNS data, the domain name to IP address mappings in the name server can be updated by a node through dynamic updates. This feature can be used by the mobile nodes to update their corresponding primary DNS whenever their IP address changes. The old IP address is deleted from the server and the new IP address is mapped with the domain name of the mobile node. In multi-homed devices updates can also be sent when one of their interfaces goes down to delete the corresponding entry from DNS.

As DNS is already widely deployed in the Internet it is a good choice to be used as a location manager. With the increase in the use of mobile devices like laptops, PDAs etc, the need to update DNS data through dynamic updates is gaining significance. On the other hand, we are putting zone data vulnerable to many security threats. Misuse of DNS dynamic updates can have many severe consequences.

DNS location updates give rise to many authentication and authorization related issues. Many inside and outside attacks are possible which can lead to denial of service. It is also thought that personal privacy can also be compromised through location tracking. The security threats related to DNS Dynamic updates need to be analyzed in order to determine the trade off between its usability and the possible consequences. The security threats once analyzed can be headed towards their solutions. In this paper we have analyzed, different security issues and the proposed solutions to minimize their effects, regarding DNS Dynamic updates.

This paper is organized as follows: section II presents the related work. Section III gives brief introduction of DNS Dynamic Updates functionality and its usability for mobile devices, and in section IV security issues related to DNS dynamic updates are discussed. Section V represents an analysis of different issues. Section VI concludes the paper.
2. RELATED WORK

Authentication and Authorization issues related to DNS dynamic updates are identified in [12]. Keeping the Key on line on a single system makes DNS vulnerable to single point of attack. Threshold cryptography can be used to solve this problem. In the proposed architecture [12], zone security servers are used which are under control of the zone manager. Using threshold cryptography zone private key is shared between primary name server which is under the control of the server administrator and the zone security servers. Since the cooperation of zone security servers is required to compute the signature of the RRs so it is difficult for the server administrator to change the signed data [12].

To avoid compromised DNS and inside attacks, a decentralized trust management system based on KeyNote2 is proposed in [4]. In the given architecture, a node sends its credentials in the additional data field every time it makes an update request. Compliance checker then processes the request and assertions to decide whether approve or reject the request.

DNS dynamic updates should be authenticated in order to protect zone data from malicious node. Authentications can be done by using DNSSEC SIG(0) or through TSI [13]. In order to authorize a node to make changes in the zone data, policies should be defined at DNS server [13].

Many security threats related to DNS and DNS transactions are identified in [7]. Their protection approaches are also discussed.

The issue of location tracking is discussed in [8]. The mobile node on its move updates the new IP addresses it acquires, against its domain name. If someone wants to track the movement of the mobile node it can send queries to DNS server to get the mobile node's new IP address. These IP addresses then can be mapped to their corresponding geographic locations.

3. DNS DYNAMIC UPDATES

Information in DNS is stored in the form of resource records (RR). Nodes can update their A (domain-name IP-address), PTR (IP-address domain-name) or CNAME (Aliases) etc. records. Four types of updates can be performed by a node: 1) Delete an RR record, 2) Add an RR record 3) Delete an RRSet (records with same name and type are called RRSet), 4) Delete all RRSet of a given name. Node can send multiple records that are to be updated, in a single message. Nodes can also specify the existence or non-existence of some records in the prerequisite section. The prerequisites must be processed first at the server before applying the updates. All the given prerequisites must be fulfilled for the updates to be applicable. Otherwise no updates are done. Update is an atomic process i.e. either all the updates in a message will be performed or none of them will be executed. There are certain types of records for which multiple records can't be added e.g. SOA (Start Of Authority) and CNAME. It is not possible for a node to have two CNAMEs. In order to add another CNAME record of a node the first must be deleted. Same is the case for SOA. SOA identify the start of the zone data and it contains the primary server address. The SOA sequence number is incremented when ever an update is performed in order to reflect changes in the zone file [10].

3.1 Updates for Mobile Devices

When a mobile node moves to a different network, its IP address changes. The domain name to IP address mapping for the node in DNS server will no longer be valid. For the other nodes to be able to send connection requests to the mobile node there must be a mechanism to locate the current position of the mobile node. Many mobility management protocols e.g. TCP-Migrate [14], TCP-Redirect (TCP-R) [5], parallel-TCP (pTCP) [6] and Tsukamoto's proposal [9] use DNS as the location management mechanism to track the current position of the mobile node. DNS dynamic updates are used to add the new IP address against the domain name in the node's authorized DNS server. Normally DHCP server updates the A and PTR records of the mobile node in the primary DNS after assigning the new IP address to the mobile node. However the mobile node can negotiate with the DHCP server, on who will send the updates.

4. SECURITY ISSUES IN DNS DYNAMIC UPDATES

Through dynamic updates different nodes are allowed to make real-time changes in the zone data therefore certain measures should be adapted to avoid their misuse. Certain security issues related to dynamic updates are listed here.

4.1 Authentication

Everyone has access to DNS server to put queries to resolve domain name to IP address mappings. Unlike DNS responses, queries need not be authenticated as they don’t put any kind of security threat to zone data. This can’t be true in case of DNS dynamic update requests. The lack of authentication requirement for dynamic update requests will put zone data vulnerable to many security attacks. Anyone will be able to manipulate zone data accordingly to perform malicious activities. This also nullifies the authentication of DNS responses. Because the zone data is open to be changed by anyone, this assurance that the response is from an authorized server is not sufficient to provide authenticity of data. Therefore authentication is the minimum requirement that must be applied to DNS dynamic updates in order to protect integrity of zone data.

Authentication based on IP address is vulnerable to IP spoofing. Therefore secure dynamic updates are used to authenticate update requests. In secure DNS TSIG and SIG(0) records are used to authenticate messages between two entities. In TSIG MAC (Message Authentication Code) is calculated through a shared secret between server and client while SIG(0) is based on public private key cryptography. The update message is signed by the host's private key whose public key is stored in DNS. The generated MAC/signature is included as the final resource record in additional data section of the update message. Authentication through SIG(0) is more scalable while calculation of MAC in TSIG is computationally expensive [13]. The shared secret can be dynamically exchanged between DNS server and the mobile node through TKEY [3]. The policies defined, based on the key used to sign the message, to authorize nodes to make changes in the zone data are fully implemented on the server [13].

Because DNSSEC is already widely deployed to authenticate DNS server response, it can also provide an efficient way to authenticate client requests. This solution does not impose any
changes in the existing infrastructure of DNS. The major overhead of this scheme is that with the increase in the number of mobile nodes DNS server will have to maintain separate unique keys for each node. One possible solution is to restrict updates only from DHCP server but this can lead to sane DHCP attacks.

4.2 Authorization

Despite authentication, update requests need to be authorized also. If the update requests are not authorized then the node once authenticated can update any resource record in the zone. Despite updating its own records it can also insert false entries regarding other nodes in the zone data. This makes DNS vulnerable to many security attacks like Redirection, Denial of Service etc. Compromising a DNS server is difficult for an attacker due to security measures implemented at the server, while less effort is needed to compromise a client system. Most client systems do not implement any security measures at all. If update requests are not authorized this opens an opportunity to the attacker to compromise a single client system authenticated with the server and have access to all the zone data to be manipulated. By calculating MAC of the update message based on the shared secret and sending it in the update message attacker can pollute the zone data with false updates.

The nodes should be given restricted access to update zone data. The policies to authorize a node to make specific updates are defined at the server. In BIND 9 [1] update-policy can be defined to restrict a node to update specific records in the zone, based on the key used to authenticate the node. For example if a node say www.example.com uses a key www.example.com then update policy can be defined as

```plaintext
zone "example.com" {
    update-policy { grant www.example.com. Self *example.com. A; }
};
```

If a single policy can't be applied to all the nodes in a zone, policies for each node will have to be defined separately. This increases the administrative overhead. Policies for DHCP servers are defined such that they are able to update zone entries for other nodes, therefore there is no way to protect zone data in case of sane DHCP servers. The above solution also does not provide any means to resist inside attacks.

4.3 Compromised DNS

Man in the middle attack was possible on DNS because DNS clients were not able to authenticate a received RR. It was possible for an attacker to intercept a query and then return a false RR in response. DNSSEC (DNS Security extensions) [2] was proposed to overcome these security concerns. In DNSSEC each RR is digitally signed by the zone private key. In response to DNS query along with the requested RR a SIG RR containing the signature of the requested RRs is also included. Resolver uses the zones public key to authenticate the received RRs [12]. Zone key can be kept off-line in case of static RRs i.e. zone data does not change frequently. Digital signatures of RRs can be computed off-line and are then put into the zone files on name server. This is feasible only in case if the zone data do not change frequently. As the mobile nodes moves across different networks their IP address changes requiring frequent changes in the zone data.

In case of dynamic updates, which allow real time changes in zone data, the changed RR will not be applicable unless it is signed by the zone key. Therefore the changed RRs have to be signed by the zone key in real time in order to be effective. Otherwise the changes will not be forwarded. For this reason, the zone key needs to be kept on line. This gives rise to many security risks such as single point of attack. If the primary server is compromised, attacker can learn the zone key. Therefore resource records signed by that key will become invalid because he can then use that key to sign malicious records in the zone [12][4]

One possible solution is to use threshold cryptography [12]. In threshold cryptography private key is divided into pieces distributed on different machines. The signature is computed by the contribution of all the pieces. In [12], architecture is proposed based on threshold cryptography. In the proposed architecture different zone security servers are introduced which are in the control of the zone manager. Zone private key is distributed among these zone security servers and the primary DNS server. As signature computation of the zone RRs requires contribution of the zone security servers therefore even if the DNS server is compromised, attacker is unable to corrupt zone data [12].

Although above solution provides a way to protect zone key but this will increase the computational overhead. Additional servers will be required by the zone managers to manage the zones. Deployment of additional servers just to protect the key will add the administrative burden and requires changes in the current infrastructure of DNS.
4.4 Inside Attacks

A single DNS server contains many zones. It is not always the case that the zone owner is also the owner of the server. By keeping the zone key on line, server administrator can have the access to the key. This could be a possible security breach as it gives rise to the possibility of server administrator abuse his/her power [12][4].

Moreover zone policies are usually defined in configuration files. These files contain keys to authenticate different nodes. Update policies based on these authentication keys, used to restrict nodes' authorizations to update any resource record of a zone are also defined in these configuration files. These file are placed on the server in readable form. Currently there are no authentication and authorization mechanisms to restrict modifications of the contents of these configuration files. As there can be thousands of zones on a single DNS server it is not possible to provide shell accounts for each zone. So it is possible for the server administrator to view configuration files of a zone. Server administrator can then modify these files allowing malicious nodes to violate zone's original policies [4].

Threshold cryptography [12], as mentioned above can be used to hide the zone private key from the server administrator. The problem of configuration files can be solved by the use of decentralized trust management system [4]. In [4] a decentralized trust management for dynamic updates based on KeyNote system is proposed to avoid inside attacks. A set of unified mechanisms are used to specify security policies and security credentials. Credentials are usually signed statements used to specify user authorizations while access rights are granted based on the public keys of the users [4].

KeyNote is a simple trust management system used to make access control decisions in a distributed environment. It has four basic components, 1) actions, these are the operations needing access control, 2) mechanisms to identify users, 3) assertions, they define which users are allowed to perform different actions, 4) compliance checker, based on a given policy and a set of credentials it is used to determine if the requested action of the given user should be allowed are denied. For DNS dynamic updates authentication of the users is done through DNSSEC authentication mechanisms. Clients provide their credentials in the additional section every time they made an update request. The compliance checker then gets the credentials of the client and action attributes from the request and assertions from the configuration file. Processes them and returns either “reject” or “approve” [4].

This solution can lead to denial of service as the time required to validate the signatures could be significant in case a large number of messages are sent to the server [4].

4.5 Replay Attacks

There is also a possibility of replay attacks. For example when a node sends an update request the attacker holds that request. The attacker can send this request to the server at a time when it is no more valid for the actual node (the IP address is not valid for the node) [7].

In Figure 1, a possible scenario of replay attacks is described. For example a node A in an example.com domain acquires an IP address and sends an update request to its primary name server. Attacker intercepts that message. During its movement mobile node A moves to another subnet and acquires new IP address, and accordingly update its entry in the primary name server. Attacker then resubmits the intercepted update message to DNS server. As the request is authenticated server updates node A's record. In this way zone data can be corrupted by adding such false entries.

Threshold cryptography [12], as mentioned above can be used to hide the zone private key from the server administrator. In this way denial of service could be launched against node A as other nodes will not be able to access node A. This can also prevent node A to make update requests if it mentions the resource record containing its previous IP address in the prerequisite section.

This attack can be prevented by using timestamp in TSIG. In this way server can terminate the update request with previously used timestamp. The time signed field in the TSIG RR contains seconds since 1-Jan-70UTC [11].

4.6 Redirection Attack

The attack in which the attacker updates its own IP address against the domain name of another node is called redirection attack. In this way attacker's IP address will be sent back in response to the queries against the compromised domain name. Redirection attacks are possible if update requests are not properly authenticated or authorization policies are not well configured.

A possible example of redirection attack is shown in Fig. 2. If the zone is not secure, anyone can change any record in zone. For example, attacker can update its own IP address against the domain name of node A. DNS will return attacker's IP address in response to the queries against node A's domain name.
Redirection attacks are also possible through sane DHCP servers or message tampering. Moreover if DNS is compromised it can be configured to send a false record to pollute resolver’s cache in order to launch Redirection attack.

To avoid redirection attacks despite authentication strict authorization policies should also be defined. This is the attack which is possible only due to certain other security issues. Therefore its prevention is based on adopting measures to avoid those security risks.

4.7 Sane DHCP Server

DHCP in a domain is authorized to make updates on behalf of other nodes. When a node acquires an IP address from DHCP, DHCP updates the corresponding entry in DNS. A sane DHCP server can manipulate the zone data accordingly to perform malicious activities.

DHCP and DNS servers should be under the same administrative body in order to have a strong trust relationship. Strong security measures should also be adopted at DHCP server in order to avoid an attacker being able to compromise the server.

4.8 Message Tampering

Update message could be tampered to insert illegitimate entries [7]. This can be solved through authentication mechanisms. Node calculates the MAC of the update message through the shared key and sends this value in the additional data section of the update message. Server recalculates the MAC to insure the integrity of the message [7].

4.9 Denial of Service Attack

All the security issues listed above can ultimately lead to denial of service attack against the nodes. There can be many different forms of this attack. It could be launched against a node by deleting its entry from DNS server. Mobile nodes can be prevented from performing updates by changing authorization policies of a zone, possible through inside attacks.

4.10 Location Tracking

Location updates also lead to privacy issues like tracking and monitoring the location of the mobile user. Whenever, the location of the mobile node changes, this change will be reflected in the DNS record by change of the IP address against the domain name. These network locations could be mapped to the geographic locations. This could be a security risk to personal privacy and business confidentiality [8].

One possible solution is to permit the node to enable or disable location updates. For example if a mobile node doesn’t want to be tracked it can turn off location updates and when it wants to provide services it can turn on this feature.

5. INTERDEPENDENCE OF DIFFERENT ATTACKS

Many attacks are interrelated to each other. The attacks which are possible due to several other attacks are summarized in Table 1 below. Denial Of Service is the ultimate attack which is possible if any one of the attacks shown in Table 1 against DOS attack is made possible by the attacker. For example in the absence of any authentication or authorization policies it is possible for any node to delete the records of other nodes. If a DNS server is compromised the attacker can manipulate the zone data to launch DoS attack. Because the key is kept on line it is possible for the administrator of the server to make changes in the zone data or change configuration files to alter update policies defined against different nodes. So, the mobile node would not be able to update its record. Denial of Service attack against a node is also possible through Message tampering, Redirection attack or through sane DHCP server.

If authentication mechanisms are not properly implemented message tampering would be possible. This could lead to many illegal operations to be performed on zone data.

<table>
<thead>
<tr>
<th>Attack</th>
<th>Possible Due To</th>
</tr>
</thead>
<tbody>
<tr>
<td>Denial Of Service Attack</td>
<td>Authentication, Authorization, Comprised DNS, Inside Attacks, Replay Attacks, Sane DHCP Server, Message Tampering, Redirection Attack</td>
</tr>
<tr>
<td>Message Tampering</td>
<td>Authentication</td>
</tr>
<tr>
<td>Redirection Attack</td>
<td>Authentication, Authorization, Sane DHCP Server, Message Tampering, Comprised DNS, Inside Attacks</td>
</tr>
<tr>
<td>Sane DHCP Server</td>
<td>Authentication, Authorization</td>
</tr>
<tr>
<td>Authentication</td>
<td>Compromised DNS, Inside Attacks, Authorization</td>
</tr>
<tr>
<td>Authorization</td>
<td>Compromised DNS, Inside Attacks</td>
</tr>
</tbody>
</table>

Table 1. Dependence of Different Attacks

Redirection attacks are possible due to many other attacks as shown in Table 1 above. Attacker can use Sane DHCP server or Message Tampering to launch Redirection attacks in case authentication mechanisms are not very effective. Redirection attacks for a domain could be launched if name server is compromised or through inside attacks.

Zone policies could be violated through sane DHCP server even if current authentication and authorization mechanisms are deployed. Therefore we need to implement more efficient solutions to prevent zone data from sane DHCP server attacks.

Authentication and Authorization policies can be modified to allow malicious records to be added in the zone data through compromised DNS or Inside attacks. Because configuration files are in readable form they can be modified by the server administrator or the attacker who have been able to compromised the DNS server.

6. CONCLUSION

DNS dynamic updates can facilitate mobile nodes for location management. However it makes zone data vulnerable to be manipulated by malicious nodes if proper security measures are not adopted. In this paper, possible security risks caused by DNS dynamic updates are summarized. Proposed solutions to prevent these security risks are also analyzed. It has been also found that most of the attacks are dependent on each other as, if some solution suggests strong measures to avoid a particular attack, that
attack may still be possible indirectly through some other attack. Therefore there is a need to develop a comprehensive solution to avoid security issues related to DNS dynamic updates. Most of these attacks are based on week authentication and authorization mechanisms. So there is a need to design some comprehensive solution that provide robust authentication and authorization services for DNS dynamic updates.

7. REFERENCES