Varun Dharmesh Vora 115098893

DNSSEC Implementation

(PART B – ASSIGNMENT 1)

1] Successful DNSSEC Verification:

For first part of B, we verify our DNSSEC resolver using multiple sites like **verisigninc.com, dnssec-tools.org, dnssec-deployment.org** and **internetsociety.org.**

The overall understanding is outlined as follows:

- We first query these websites to the root servers we fetched from https://www.iana.org/domains/root/servers
- We then obtain the DNS and DNSKey response for this website which is further used to verify the authenticity of the PubZSK and DS records.
- The root is then verified using the custom verify function and we then resolve the query ahead to the servers we fetch from the additional section of root.
- The verify function is the validation function that validates the dnssec(using three different verifications in order to secure the responses from man of the middle attack) and according to its response, we move ahead to resolve the next query.
- The first verification we perform is the verification of the zone by verifying the PubKSK using the hashing algorithm. We store the DS received from the parent website of the current website and compare it to the hashed value of the PubKSK we received from the current website. If the hashed value matches to the stored DS, the PubKSK is verified. This helps us transfer the trust of the parent website to the child website building a chain of trust between these websites.
- The second verification we perform is to verify the RRSig of the DNSKey RRset by using the PubKSK.
- The third verification we perform is to verify the RRSig of the requested RRset by using the PubZSK since it was signed by the PvtZSK(using the public-private key methodology).
- These 3 verification steps are repeated for every iterative query to build a chain of trust between the servers and their responses.
- After iterating through the name servers in the hierarchy, we finally reach the Authoritative Name Server which contains the A records which we can use to resolve the IP of the domain. Here, we handle this using the veriA function in our code where instead of DS records, we get A and it's corresponding RRSig in the answer section of the response.
- After the verification of the final step, we then resolve it to get the final IP address.

2] Unsuccessful DNSSEC Verification:

There are two cases where the DNSSEC Verification fails which are **DNSSEC not supported** and **DNSSEC Verification Failed**

a) **DNSSEC not supported**

- The example used for this case is **google.com.**
- This case is handled using the concept of NSEC3 and NSEC flags. Every time we get a response for the type A query, we check whether there exists a NSEC/NSEC3 record. It's existence provides us with an

authenticated denial of existence. Using this we conclude that this particular website does not support DNSSEC.

b) **DNSSEC Verification Failed**

- The examples that can be used for this case are **dnssec-failed.org** and **rhybar.cz.**
- There are three verifications that need to be passed in order to validate the query response. In our examples, for dnssec-failed.org the PubKSK is not being verified after comparing it to the DS record from the parent zone. Similarly for rhybar.cz, the A and it's corresponding RRSig are not getting verified. Hence, the DNSSEC Verifications fail.