**DNSSEC implementation**

**Objective**: Implement DNSSEC as accurately as possible

**Implementation:**

For each query, the following steps are implemented:

1. Get the desired RRSET using a regular query, but with the DO flag set.
2. We get the RRSET desired along with a Delegation Signer (DS) and Digital Signature(RRSIG).
3. Query for the DNSKEY
4. We get the DNSKEY and the RRSIG of the DNSKEY  
   **Verification**
5. Calculate the DS for the queried DNSKEY.
6. Verify that the DS is the same that was sent in the parent record. This verifies the “chain-of-trust”

**Special processing for the root:**

* Since we do not have a “parent” for the root, the DNSKEY of the parent is verified against an “offline” set of Public keys. These are provided by the OS and refreshed periodically. Assumption made**:** The DNSKEY of the root server is usually bootstrapped and periodically refreshed to keep them up to date. The implementation here does not do that for the sake of simplicity (and possible invalidation after a certain period).

**Special processing for the “last record” or AA: Authoritative Answer**

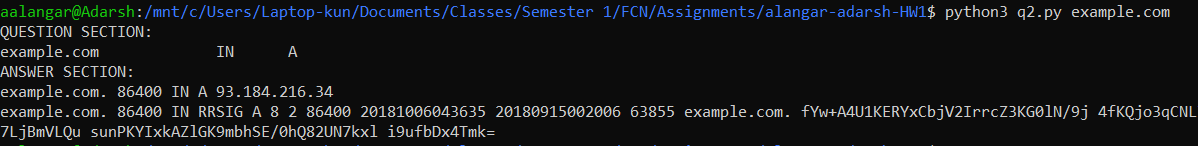
* For the AA record, we also need to check the validity of the answer section.
* To do this, we check the RRSIG of the A record against the A record and the public DNSKEY (Zone Signing Key).
* Since the DNSKEY has been validated using the parent DS, if the previous validation passes, we know that the IP returned is valid and authentic.

**Checking Invalid and DNSSEC not implemented**

* **Invalid DNS:** As seen in the implementation, if the validation fails at any point, the program exits with an error: “DNSSec verification failed”
* **DNSNEC not implemented:**
  + If DNSSEC is not implemented by a child server, the parent responds with a NSEC RR.
  + If the NSEC RR contains DNSKEY or RRSIG, this informs the resolver that the child domain does not support DNSSEC
  + At this point we exit with the error: DNSSEC not supported

**Output:**

**Valid Resolution:**



**DNSSEC not supported:**



**DNSSEC verification failed:**

