Recitation 23 — DNSSEC

Goal of DNSSEC

- Make sure information in DNS response came from a legitimate server
- Maintain backwards compatibility with DNS
- Goal is *not* to provide confidentiality for responses. We're not worried about attackers being able to read the responses from a server to a client; we're worried about servers lying.

Building up to DNSSEC

Why doesn't DNSSEC use a more straightforward application of public-key cryptography?

- Naive application of public-key crypto to DNS: Client makes a request to server S, S responds with an encrypted response, client decrypts with S's public key.
    - Extend this to having every name server that a client needs to contact do this.
    - Downside: method isn't backward-compatible with current DNS (clients that don't use DNSSEC still have to decrypt responses)
- Second idea: Use signatures. non-DNSSEC zones/clients/etc. can ignore signatures. Signature is in new resource record, RRSIG.
    - Problem: how to distribute keys? Centralized authority defeats the distributed nature of DNS.
    - Solution: add a new resource record, DNSKEY
    - RRSIG needs to contain a hash of the original contents so that an on-path attacker can't change its data
- Attack: third party inserts a response that changes the RRSIG *and* DNSKEY records. Occurs because we haven't authenticated the DNSKEY resource record.
- Solution to this attack: chain of trust starting at the root. Parent authenticates its child's DNSKEY via the DS resource record

Discussion

- Why hasn't DNSSEC been fully deployed? Possible reasons:
    - It's complex to implement/set up
    - DNS seems to be working fine as is
    - Little motivation to deploy if only a few zones are signed
    - People are worried about zone enumeration attacks
- Who should be in charge of the root key?