# Network Security II

**Question 1** ***TLS threats*** ()

An attacker is trying to attack the company Boogle and its users. Assume that users always visit Boogle's website with an HTTPS connection, using ephemeral Diffie-Hellman. You should also assume that Boogle does not use certificate pinning. The attacker may have one of three possible goals:

1. Impersonate the Boogle web server to a user
2. Discover some of the plaintext of data sent during a past connection between a user and Boogle's website
3. Replay data that a user previously sent to the Boogle server over a prior HTTPS connection

For each of the following scenarios, describe if and how the attacker can achieve each goal.

(a) The attacker obtains a copy of Boogle's certificate.

> **Solution:** None of the above. The certificate is public. Anyone can obtain a copy simply by connecting to Boogle's webserver. So learning the certificate doesn't help the attacker.

(b) The attacker obtains the private key of a certificate authority trusted by users of Boogle.

> **Solution:**
>
> The attacker can impersonate the Boogle web server to a user. The attacker can't decrypt past data. First, Boogle's private key is used in the protocol, not the CA's. Second, Diffie–Hellman provides "forward-secrecy" (as in part (c)), and so the attacker could not decrypt it regardless.
>
> The CA's private key can be used for creating bogus certificates, which can be used to fool the client into thinking it is talking to Boogle.
>
> Replays aren't possible, due to the nonces in the TLS handshake.

(c) The attacker obtains the private key corresponding to an old certificate used by Boogle's server during a past connection between a victim and Boogle's server.

Assume that this old certificate has been revoked and is no longer valid. Note that the attacker does not have the private key corresponding to current certificate.

**Solution:** None. Unless the attacker can figure out either a or b, the attacker will not be able to decrypt the data of past connections.

This can't be used to impersonate a Boogle server because the attacker doesn't have a fresh valid certificate corresponding to the stolen private key, and can't use the previous certificate for that key because it's been revoked.

**Question 2** ***DNS Walkthrough*** ()

Your computer sends a DNS request for "www.google.com"

(a) Assume the DNS resolver receives back the following reply:

```
com. NS a.gtld-servers.net
a.gtld-servers.net A 192.5.6.30
```

Describe what this reply means and where the DNS resolver would look next.

> **Solution:** The IP address for "www.google.com" is not known. However, "a.gtld-servers.net" is a name server for .com, and that is where the resolver should ask next, at the IP address 192.5.6.30.

(b) If an off-path adversary wants to poison the DNS cache, what values does the adversary need to guess?

> **Solution:** The adversary will need to guess the identification number (16 bits). Some resolvers even randomize source ports.
>
> The reason an off-path attack is difficult is because the ID (and port numbers) have to match exactly, but once the legitimate reply reaches the resolver and is cached, the server is no longer vulnerable to the poisoning attempts.

(c) Why not use cryptography to make the DNS connection secure?

> **Solution:** DNS is designed to be lightweight and cryptography (eg. TLS) adds a lot of overhead.
>
> Furthermore, we do not know which name servers to trust and TLS provides no protection against that. This is a fundamental difference between object security and channel security.
>
> There is a DNS-over-HTTPS protocol that can be used today and is becoming increasingly popular.

## Question 3 *DNS* (14 min)

(a) Alice wants to access Berkeley's diversity advancement project DARE, `dare.berkeley.edu`. Her laptop connects to a wireless access point (AP).

Alice worries that a hacker attacks the DNS protocol when her laptop is looking for the IP address of `dare.berkeley.edu`. Assume that DNSSEC is not in use.

◇ **Question:** Which of the following can attack the DNS protocol and have Alice's browser obtain an incorrect IP address for DARE? (Select 0 to 8 options.)

- ■ The laptop's operating system.
- ■ The laptop's network interface controller.
- ■ The wireless access point.
- ■ An on-path attacker on the local network.
- ■ The local DNS resolver of the network.
- ■ The root DNS servers.
- ■ `berkeley.edu`'s DNS nameservers.
- ■ An on-path attacker between the local DNS resolver and the rest of the Internet.

**Solution:** Anything that can spoof a DNS response from the resolver or a nameserver. In this case, all of these options have that capability.

(b) Now assume that `berkeley.edu` implements DNSSEC and Alice's recursive resolver (but not her client) validates DNSSEC.

◇ **Question:** Which of the following can attack the DNS protocol and have Alice's browser obtain an incorrect IP address for DARE? (Select 0 to 8 options.)

- ■ The laptop's operating system.
- ■ The laptop's network interface controller.
- ■ The wireless access point.
- ■ An on-path attacker on the local network.
- ■ The local DNS resolver of the network.
- ■ The root DNS servers.
- ■ `berkeley.edu`'s DNS nameservers.
- □ An on-path attacker between the local DNS resolver and the rest of the Internet.

**Solution:** Any on-path attacker can see DNS traffic and spoof responses from the resolver. The on-path attacker between the resolver and nameservers can't spoof any nameserver responses because of DNSSEC.

(c) An attacker wants to poison the local DNS resolver's cache using the Kaminsky attack. We assume that the resolver does not use source port randomization, so the attacker will likely succeed.

In the Kaminsky attack, the attacker asks the resolver for a *non-existing* subdomain of UC Berkeley, *e.g.*, `stanford.berkeley.edu`, instead of asking for an *existing* domain like `dare.berkeley.edu`.

◇ **Question:** What is the advantage of asking for a non-existent domain compared to asking for an existing domain? (answer within 10 words)

________________________________________________________________

________________________________________________________________

**Solution:** When you fail, you can keep trying with another nonexistant name/race until win!

(Note, caching alone is not sufficient, because you do have caching of NXDOMAIN too. The big thing is "race until win". (3 points))