Public Key Infrastructure

24.1. The following is an X.509 certificate.

```
Certificate:
 Data:
   Version: 3 (0x2)
   Serial Number:
     3d:0e:98:b2:bf:af:fa:9e:99:91:05:64:69:6e:11:2a
 Signature Algorithm: sha256WithRSAEncryption
   Issuer: C=US, O=Symantec Corporation,
           OU=Symantec Trust Network,
           CN=Symantec Class 3 EV SSL CA - G3
   Validity
      Not Before: Aug 14 00:00:00 2017 GMT
      Not After : Sep 13 23:59:59 2018 GMT
   Subject: ... C=US/postalCode=22230, ST=Virginia,
         L=Arlington/street=4201 Wilson Blvd,
         O=National Science Foundation, OU=DIS,
         CN=www.nsf.gov
   Subject Public Key Info:
     Public Key Algorithm: rsaEncryption
       Public-Key: (2048 bit)
         Modulus:
            00:ca:fb:26:78:06:25:b1:9e:67:1d:69:0b:10:06:
            cf:25:b6:7d:de:8e:56:80:e1:1c:38:52:62:43:fd:
            ...
      Exponent: 65537 (0x10001)
 Signature Algorithm: sha256WithRSAEncryption
     4b:0d:62:11:b4:dc:78:09:12:c1:1b:24:ff:98:43:58:1c:54:
     0a:34:be:8f:3f:12:8f:17:4a:fe:5b:26:13:1a:5f:a7:87:ad:
     ...
     ba:2c:10:c7:bc:8b:2c:15:6e:0c:d2:d0:8b:74:52:c8:ed:05:
     0b:9b:62:41
```

(a) Who issues the certificate?

(b) Who is the owner of the certificate?

(c) Who generated the signature on this certificate, and how can this signature be verified?

(d) The public key contained in this certificate is based on the RSA algorithm. Using the RSA algorithm, to encrypt a message M, we calculate M^e `mod n`. What is the value of `e` and `n` in this public key? If a number is too large, you only need to write down its first four bytes.

(e) Before issuing the certificate, the CA needs to do a verification regarding the subject field. Please describe what this verification is, and why it is necessary.

24.2. When browsing a web site, we see the following message. What does it mean that the certificate is not issued by a trusted CA? What is considered as a trusted CA?

```
There is a problem with this website's security certificate.
The security certificate presented by this website was not
issued by a trusted certificate authority.
```

24.3. A bank recently changed its website name from `www.bank32.com` to `www.bank48.com`, so users have to use this new name to access the bank's online services. To cut the cost, the bank wants to use the same certificate, instead of getting a new one. Would that be possible and why?

24.4. An attacker has created a self-signed certificate, and he somehow gets a victim to add this certificate to the trusted certificate list of the victim's browser. What could be the damage?

24.5. Find 3 sites that use EV certificates.

24.6. Instead of typing `https://www.example.com` in the URL field of a browser, we first get the IP address of the web server, which is `93.184.216.34`, and we then directly type `https://93.184.216.34` into the browser. Describe whether we will be able to connect to the web server.

24.7. If a CA's private key is stolen by an attacker, what damages can the attacker achieve?

24.8. Please explain what certificate pinning is, and what it can achieve.

24.9. We know that HTTPS can defeat man-in-the-middle attacks. However, we also know that HTTPS proxy can be installed to monitor and modify HTTPS traffic. A proxy is basically a "man" in the middle. Does this mean that HTTPS is still subject to man-in-the-middle attacks? Please explain.

24.10. In my class, students need to implement a simple VPN program, in which the client and server communicates via SSL, i.e., the communication between them are encrypted. To connect the VPN client to the VPN server, most students choose to ask users to type the IP address of the VPN server on the command line. To compare with the common name in the server's certificate, these students also ask users to type the intended common name (i.e. the host name of the server) at the command line. Although this practice is not good from the usability perspective, it has no harm.

One student decides to improve the usability, and he wants avoid asking users to type the host name. He did a reverse DNS lookup using the IP address provided, so he can get the hostname from the lookup. He then compare the host name with the common name in the certificate; if it matches, the server is trusted.

Please answer the following questions. We will use `syr.edu` as an example. Assume that `syr.edu`'s VPN server is `vpn.syr.edu` with IP address `128.230.53.1`. Therefore, our command will be "`vpnclient -s 128.230.53.1`". We assume that no CA is compromised, so you cannot get a fake certificate.

(a) Please describe an attack that can fool the VPN client to accept your certificate with a common name `attacker32.com`. If your certificate is accepted and the vpn client is connecting to you, you can get the user's account credentials.

(b) If we use "`vpnclient -s vpn.syr.edu`", instead of the IP address, we will use a forward DNS lookup to find the IP address. Explain whether we have the same vulnerability as that in the question above.

24.11. To authenticate a VPN server, one thing we need to do is to check the common name. To do that, instead of using `strcmp()`, a developer decides to use `strstr()`. Here is the modified comparison:

```
if (strstr(commonName,"vpnserver.com")==NULL)
{
     printf("names do no match\n"); exit(-1);
}
```

The idea is to relax the match, so the match is not restricted to `vpnserver.com`; even `abc.vpnserver.com` is also a match. This way, one site only needs one certificate. Exact match requires each URL to have a certificate, which is definitely not practical. The explanation of `strstr()` is the following:

```
char * strstr(char * str1, char * str2):
   Return a pointer to the first occurrence in str1
   of the entire sequence of characters specified in str2,
   or a null pointer if the sequence is not present in str1.
```

Is such a relaxation secure or not? Please explain.