CSEP 590TU Assignment #8 – Certificates & PKI

Due at the beginning of class on February 28, 2006

Fun with Revocation: In this problem we're going to explore some of the performance characteristics of CRLs and OCSP responses. We're going to base our numbers on the current VeriSign CRL for SSL server certificates; you can find all of VeriSign's CRLs at http://crl.verisign.com/; the one we're interested in is the RSASecureServer.crl file. This CRL is (as of 3am Wed., Feb. 22) valid from 2/22/06 to 3/8/06, is 515,243 bytes in size, and has 14,714 entries in it. Assume that all of the certs listed on the CRL were issued within the past 12 months. VeriSign claims to have about 500,000 sites with "Secure Server IDs", so assume that's the universe from which 14,714 certs have been revoked.

Question 1(a): Assume that there are 200,000,000 users who will negotiate an SSL/TLS session with at least one of the 500,000 sites with "Secure Server IDs" over the next two weeks. On average, how much bandwidth is VeriSign going to use per day distributing the RSASecureServer CRL? (You may assume user requests for CRLs are evenly distributed throughout the CRL's two-week validity period.)

Question 1(b): Now assume that VeriSign also makes its revocation information available via an OCSP responder service. If the average size of an OCSP request/response message pair is 3KB, how many OCSP responses would the average user have to request from the VeriSign OCSP responder per day in order to generate the same about of bandwidth usage as the CRL downloading you calculated in Question 1(a)?

Question 1(c): Now let's suppose that the US Government wants to investigate the feasibility of issuing a certificate to every citizen that holds a US passport (approximately 60 million people). Based on our assumptions above, VeriSign is experiencing about a 3% revocation rate for their Secure Server IDs. Let's assume that the same rate would apply for certificates issued to US passport holders. Approximately how big would the CRL be for the personal certs issued by the US Government? You may assume that each CRL entry requires 35 bytes of storage when ASN.1 encoded.

Auto-enrollment: One of the key problems for enterprises that wish to deploy PKIs is the task of initially enrolling all of their users for certificates. Suppose that you are an IT administrator within a 100,000-user enterprise and your CIO says that you need to deploy a PKI and enroll every user for two S/MIME certificates (one for their encryption key and one for their digital signature key). Your users currently authenticate using Kerberos with passwords (you have one Kerberos realm for all 100,000 users).

Question 2: Design a certificate enrollment protocol for enrolling each user for their two certificates that leverages the user's Kerberos credentials to authenticate the certificate requests to the CA. You can choose whether users enroll for both signing and encryption certificates simultaneously (in one execution of the protocol) or sequentially (in two executions of the protocol).

Question 3: After coming up with your initial design, your CIO informs you of an additional requirement: encryption key escrow at the CA. Users are required to submit a copy of their private encryption key to the CA in order to get their public encryption key certified. Modify the protocol you design in Question 2(a) to include a key escrow feature for the encryption key pair. (Users can still enroll for signing certificates simply based on their Kerberos credentials, but now the encryption certificate requires both Kerberos credentials and private key deposit.)

Certificate Renewal and Rollover: When a certificate is about the expire, the subject of that certificate often wants to *renew* the certificate with the issuing CA, keeping the name-key binding but updating the validity period in the new certificate. Alternatively, the subject might decide that it's time to generate a new key pair and request that the old name-key binding in the expiring certificate be *rolled over* into a new binding between the same name and the new key. Figure 1 depicts the renewal and rollover scenarios at the end-entity certificate level.

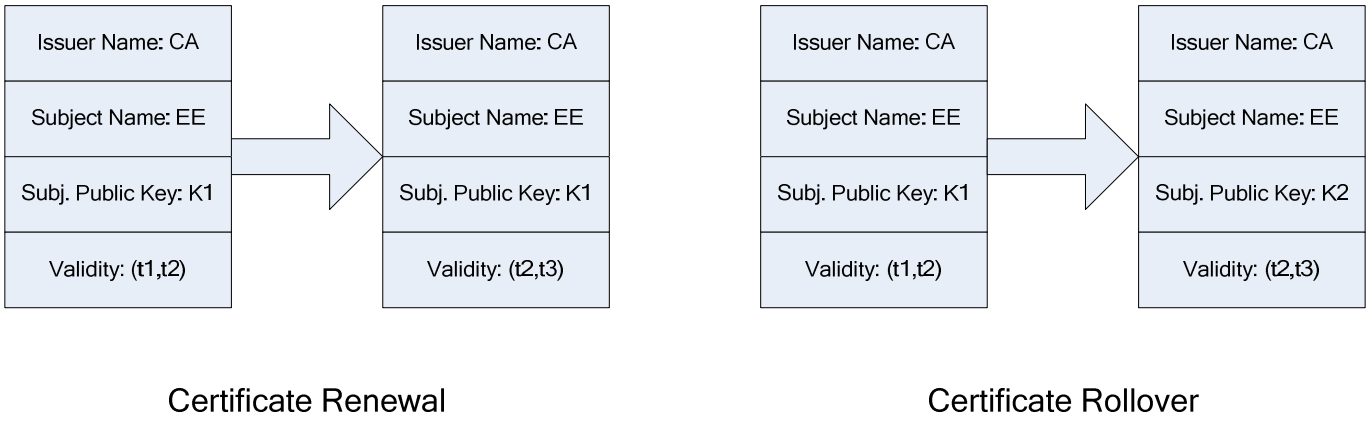

Figure 1: Certificate Renewal and Rollover

Now consider what happens when an intermediate CA has to perform a certificate renewal or rollover operation. In the renewal case, certificates issued by the CA before and after the renewal will continue to chain properly, as shown in Figure 2.

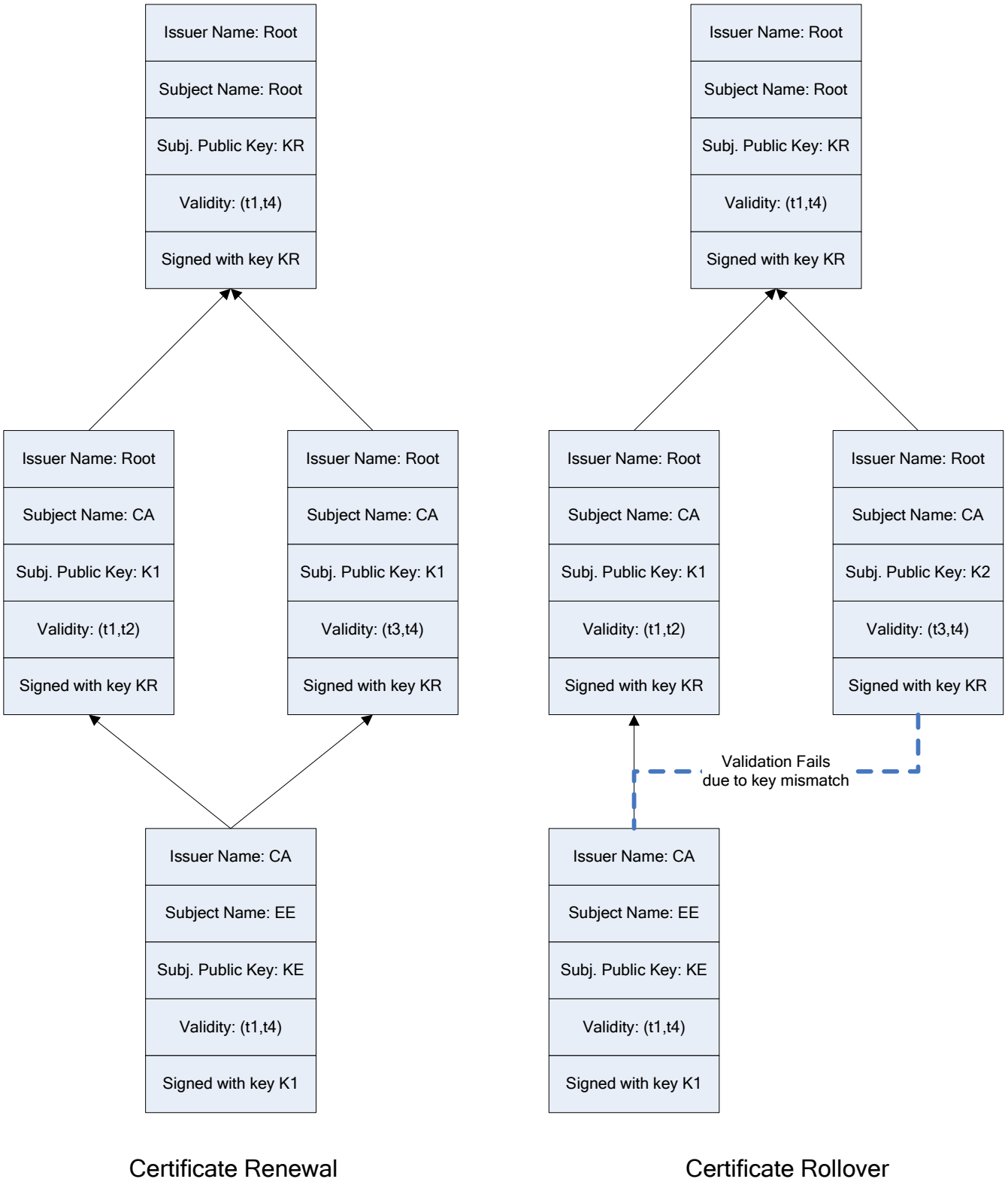

Figure 2: Intermediate CA renewal and rollover

In the CA rollover case, end entity certificates issued under the old intermediate CA certificate fail to validate under the new CA certificate because the CA's new subject public key didn't sign the end entity certificates.

Question 4(a): Assume that $t1 < t2 < t3 < t4$ and that at the time of the rollover the intermediate CA possesses both key pairs associated with public keys K1 and K2. What can the CA do to make the end-entity certificates validate at times $t3 < t < t4$ without re-issuing all of the end-entity certificates? [Hint: only one additional certificate is required.]

Question 4(b): Now assume that $t1 < t3 < t2 < t3$; that is, there is a period of overlap where both the "old" and the "new" intermediate CA certificates will be valid. As of time t3 the CA will begin issuing certificates using the new key K2. For the period of time $t3 < t < t2$ end entity certificate should be able to chain-validate under both the old and new intermediate certificates. Extend your solution to Question 4(a) to show how the CA can enable seamless rollover during the transition period.