Assignment #7 – Solutions

Problem 1a

- Migrating secrets between platform configurations on a single machine.

- TPM command to seal S:
  $C = Seal(K_{EPUB,A}, S, PCR_0, PCR_1, PCR_2, PCR_3, PCR_4, PCR_5)$

- TPM command to unseal C and retrieve S:
  $Unseal(K_{EPRIV,A}, C, PCR_0, PCR_1, PCR_2, PCR_3, PCR_4, PCR_5)$

Problem 1b

- Patching part of the measured OS
- Before applying the patch, the OS needs to seal S to the set of *new* PCR values that match the patched OS and save this sealed value away
  - Then, after patching, the PCR values of the patched OS will update and allow this sealed value to be unsealed.

$$Seal(K_{EPUB,A}, S, PCR_0, PCR_1, PCR_2, PCR_3, PCR'_4, PCR'_5)$$

Problem 2

- Migrating secrets between machines with TPMs
- Basic idea: use $Seal$ (without any PCR values) to encrypt S to B's TPM.
- A needs a copy of B's public encryption key for $Seal$, so
  - A queries the PKI (or perhaps B itself) for a copy of the PKI'scertificate issued to $K_{EPUB,B}$. A verifies that the cert chains properly to the PKI's trusted root, then extracts $K_{EPUB,B}$ from the certificate
  - With $K_{EPUB,B}$, A computes
$$C = Seal(K_{EPUB,B}, S)$$
  - A then sends C to B
  - B computes $Unseal(K_{KPRIV,B}, C) = S$.

Problem 3

- Migrating secrets between machines with TPMs with assurance of equivalent software stacks
  a) Before sending S to B (in any form), A should receive from B some proof that B's OS boot sequence (measured by PCRs 0-5) is the same as A's.
  b) When S is transmitted to B, A should ensure that S will only be accessible on B if the same software stack is running on B as on A at the time the migration begins.

Problem 3a

- "Before sending S to B (in any form), A should receive from B some proof that B's OS boot sequence (measured by PCRs 0-5) is the same as A's."
- A can request an attestation from B to something that includes PCRs 0-5.
  - To make sure the attestation is "fresh", A should send a random challenge value to B and ask B to $Quote$ that.
- $A \rightarrow B$: random value R (say 128 bits of random)
- $B \rightarrow A$: $Q = Quote(K_{SSIG,B}, R, PCR_0, PCR_1, \ldots, PCR_5)$
- A verifies the signature on Q, checks R hasn't changed, checks $PCR_0, \ldots, PCR_5$.

Problem 3b

- "When S is transmitted to B, A should ensure that S will only be accessible on B if the same software stack is running on B as on A at the time the migration begins."
- A uses $Seal$ to encrypt S to B and the values of PCRs 0-5
  - We assume A has a copy of $B's$ public encryption key $K_{EPUB,B}$
- Using $K_{EPUB,B}$, A computes

$$C = Seal(K_{EPUB,B}, S, PCR_0, PCR_1, \dots, PCR_5)$$

- A then sends C to B
- B computes $Unseal\left(K_{KPRIV,B}, C, PCR_0, PCR_1, \dots, PCR_5\right) = S.$