Crytography and Tera TVMM

Public Key Cryptography

-Symmetric key cryptography is when two people use same key to encrypt and decrypt a message.
-To break that, each person has a public key, which can be given to anyone, and a private key, which is kept secret.

Alice P_A – Public Key
S_A – private Key

To encrypt a message to Alice

$C = E(P_A, M)$

Cipher text encrypt message

To read message

$M = D(S_A, C)$

Decrypt

-PA = pad lock
-SA = key
-Send safe with pad lock encrypt or e put message in safe.

RSA

To generate public / private key Alice does

1. Pick primes p, q
2. $N = p * q$, $P(N) = (p - 1)(q - 1)$
3. Find e and d s.t. $(e * d) = 1 \bmod P(N)$
4. $P_A = (e, N)$
   $S_A = (d, N)$

RSA Encryption and Decryption

$C = E((e, N), M) = M^e \bmod N$
$M = D((d, N), C) = C^d \bmod N$
$= M^{ed} \bmod N$
(Fermat's Little Theorem) = M

N = 35 , P = 7 , P (N) = 24, q = 5 , e = 7 , d = 7 in real systems e and d different

$P_A = (7, 35)$
$S_A = (7, 35)$

$E(P_A, 2) = 2^7 \bmod 35 = 23$

$D (S_A , 23) = 23^7 \bmod 35 = 2$

Public key signatures

Each user has private signing key and a public verification key, S_A and P_A respectively.
$S = Sig (S_A , M)$
$Verify (P_A , M , S) = Valid \text{ or } Invalid$

To generate public / private key Alice does

1. Pick primes p, q
2. $N = p * q , P (N) = (p – 1)(q – 1)$
3. Find e and d s.t. $(e * d) = 1 \bmod P (N)$
4. $P_A = (e , N)$
   $S_A = (d , N)$

$Sig ((d , N) , M) = M^d \bmod N$
$Verify ((e , N) , M , S) = Valid \text{ iff } S^e \equiv M \bmod N$

Signing Long Messages

-Use hash function
$S = Sig (S_A , h (m))$
↑
hash function

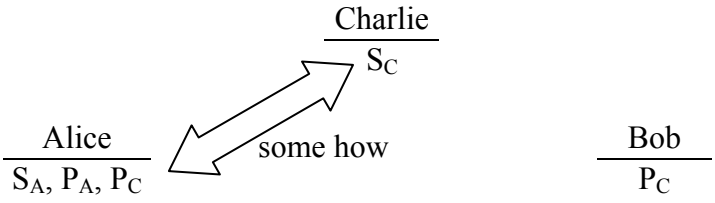
$h : \{ 0 , 1 \}^* \text{ binary string of any length} \longrightarrow \text{to } \{ 0 , 1 \}^n$

Strong Collision Resistance

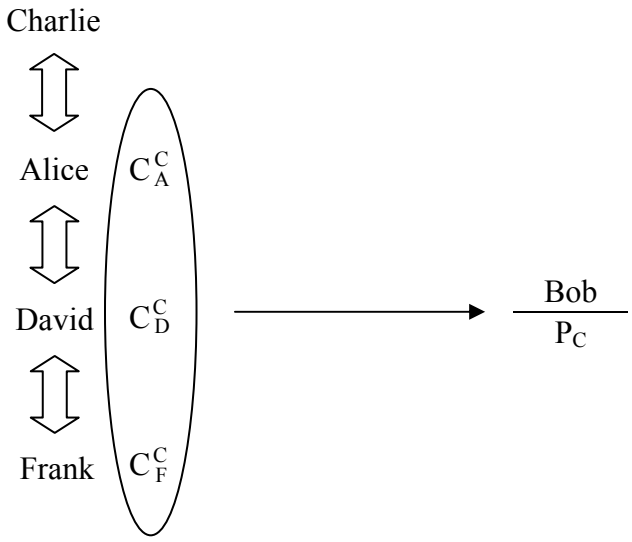
h is S.C.R. if it is hard to find $x \neq y$ such that $h (x) = h (y)$

Certificates

To verify Alice's signature, Bob needs to know P_A. Suppose Alice and Bob trust Charlie



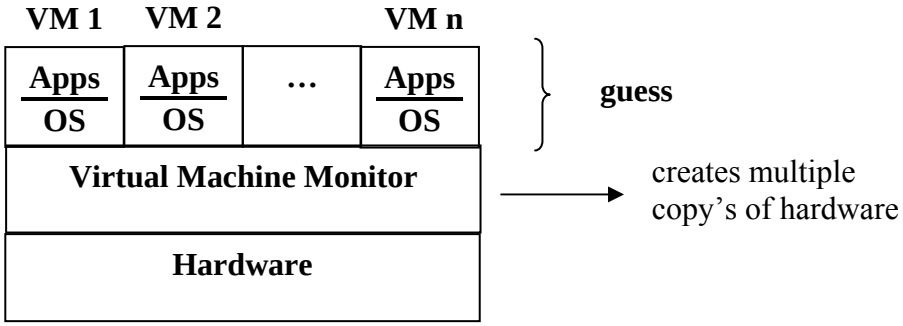
$C_A^C = (Sig (S_C, \text{“ Alice public key in } P_A \text{”}))$



Terra, TVMM

-The owner of the computer is malicious
-Need "root-secure" system
-Applications
  -Network Games
  -Movie/Music Players (aka Digital rights Management (DRM))
  -Reverse Engineering

Trusted Virtual Machine Monitor



-VMM exposes hardware interface to guest VMs
-strong isolation between VMs
  -memory
  -CPU
  -Disk
  -Network, etc

Remote Attestation

Goal: Prove to a remote party that I am running a certain set of software: App

OS
VMM
Boot Loader
BIOS

If attacker can circumvent any part of those software layers then it can fool other system in believing that it is running a certain set of software.