< Rough Notes>

Alice 128-bit secret key: secret key should be long lasting, encryption, it shouldn’t use to authenticate, Alice should be the only one who knows the secret key

Right now, Alice is using the secret key she is sending to Bob to make the HMAC. However, even if someone else makes a random secret key and uses that to make the HMAC, when Bob takes that secret key to make the HMAC again to see the authentication, Bob will not know that it is a malicious attacker. Therefore, the key used to make the HMAC should not be done using the secret key being sent, but it should be a secret key that has been predetermined. The authentication process becomes meaningless even if HMAC is used.

Bob should ask a CA to sign his public 256-bit RSA key using CA’s private signature key, not the 256bit ECC key. The ECC key is a PKE and is only used for encryption. A public verification key should only sign private signature key.

RSA keys are short lived.

Needs to send a verification key, RSA keys are not used.

Bob CA public verification key, but he is getting it in an RSA key,

Public RSA key 를 public root ca로 encrypt 로 해

Send public RSA key 를

CA한테 사인해주세요 하고 보내는게 RSA key 가 아니라 public verificaoint key

RSA는 public encryption key,

rSA key 는 short lived, public verification key는 한 번 받으면 계속

RSA key 가 verification key가 될 수 없다. Public verification key allows to be decrypted.

Need to send the public verification key, but sent the RSA key

Assume no root ca, this could be right? Exclude the verification step. CA가 문제니깐

Public rsa key, it is public anyways, does it need? Rsa

Other scenarios

(a) [4 points] Alice wants to communicate with Bob, her friend, securely. She generates a public/private key pair using RSA (128 bits), sends Bob the public key in person, and then Bob uses the public key to encrypt secret keys for AES (128 bits) in CTR mode and sends them to Alice.

Solution: 128-bit RSA is not secure because a brute force factorization attack can break 128-bit RSA quickly. RSA should be at the bare minimum 1024-bit, or better 2048- or 4096- bit.

(b) [4 points] To store passwords securely, a website administrator uses AES encryption with a secret key and a 64-bit IV to encrypt all users’ passwords. A CRC32 checksum is used to ensure correctness against random bit ﬂip errors.

Solution: Encryption should not be used to store passwords. The secret key could be stolen alongside the password database. It is better to use a hash.

(c) [4 points] A website has a TLS certiﬁcate, which is a CA’s RSA signature of the website’s private ECC key. After the visiting web user veriﬁes the ECC key, all further communication between the client and the server is encrypted and decrypted with this key.

Solution: The CA should sign the website’s public ECC key, not the private key. Otherwise, the web user cannot verify it unless she also acquires a copy of the private ECC key. (3 marks at most for ECC not being eﬃcient to encrypt and decrypt.)

(a) 16,226,837,333, 1,789,118,085 Gbit/s respectively. The former is almost 10 times more. Many people do not want to be exits because this may make them legally liable for the traﬃc that they carry out, and they may be banned by their ISPs.

(b) 5420, 30920 median ms to load respectively for 50 KiB and 5 MiB. Median download rate is 51200/5420 = 9446 B/s for the former and 5242880/30920 = 169560 B/s for the latter. The large diﬀerence is because the total download time includes the latency of the ﬁle. The amount of time required to download a small ﬁle is dominated by the latency, whereas the amount of time required to download a large ﬁle is dominated by the actual download rate.

(c) Disadvantage: Latency is higher with three nodes. Advantage: If we use one node, it can see both our true source and destination, so we must trust it (and we should not trust it because it is a volunteer). This compromises our privacy

Installed themselves as a trusted certificate provider. Signing keys on the fly. Private key is sitting on your computer and can be extracted. Same on every single computer. If one computer can pull it off one computer, every single installation is vulnerable, b/c every single computer that has superfish trusts superfish.

