Applied Cryptography

Cryptography fundamentals for a developer

Boney Johns
19/03/2019

Topics

- Hashing
- Symmetric Encryption
- Asymmetric Encryption
- SSL
- SSH
- Digital Signatures

Hashing

- One way encryption (Non reversible)
- Use case: Storage of passwords
- Common algorithms: SHA256

Symmetric Encryption

- Single key used for encryption and decryption
- 1000 times faster than Asymmetric encryption
- Main block of SSL, SSH

Asymmetric Encryption

- Key Pair - Public key & Private key
- Usually private key resides with the server and public key will be distributed
- Usually public key will be used for encryption and private key for decryption (except for digital signatures)
- Boot loader for SSL, SSH

SSL

- CA (Certificate Authority) will sign the server certificate
- Certificate contains the public key & passed to client
- Private key with server
- Symmetric session key will be generated by browser, encrypted using public key and passed to server
- Symmetric session key used to encrypt traffic in subsequent calls

SSH

- ssh-keygen - will generate key pair
- Known hosts (ssh-keyscan)
- First create the encryption channel (using seperate key pairs at client side & server side)
- Then authenticate the user (password/SSH key) via the encryption channel
- Use cases: Remote login to the server (SSH enabled), get code from Git repository
- In depth: https://www.digitalocean.com/community/tutorials/understanding-the-ssh-encryption-and-connection-process

Digital Signatures

- To verify someone is whom they claim to be
- Private key and public key is with the sender
- Create signature using hash of commonly agreed data, signed with private key
- Pass signature and public key to the receiver
- Receiver verifies the identity of the sender
- Use cases: Hub provisioning, AWS SNS message verification, .NET Dll verification

Demo

- https://bitbucket.org/bnyjohns/cryptography/src/master/

Thank You