

Security

Goal #1 — Secrecy

Insecure Channel

Bad guys listen in

e.g. credit card

e.g. password sniffing

Solution — Encryption

1. Old "Shared Secret" (WWII)

Both have secret key, k_1

Encrypt w/ k_1 , Send, Decrypt w/ k_1

Problem: key distribution

2. New "Public Key"

1. Bring up insecure connection

2. Use public key / key exchange

Public Key technology allows you to set up encryption for the dialog without a previous shared secret.

It's an amazing feat of mathematics that this can be done

Rules...

1. Transmission - SSL - HTTPs

Use encrypted transmission for secrets

Cost

SSL connections are more costly (slow) to bring up

2. Storage

Don't store a secret in a cookie or in HTML sent to client

Don't store credit-card numbers in the clear in your server DB -- encrypt them, and then concentrate on being careful with the key.

3. Virus / Worm

Don't run untrusted code so that it has access to your secrets.

Goal #2 – Authenticity / Login

Verify identify

1. Passwords

Incoming connection

Get password

Compare to local -> "authenticated"

Problem: Spoofing - Posing

e.g. fake ATM booth

e.g. fake hotmail login

Digital Signature Tech

Signed by A

B gets a copy

B verifies A's sig

1. Only A could do the signing
2. The sig was on this exact doc

Certificate Authority - CA

Issue certificates

Expensive

Currently this is artificially expensive because there are not very many CAs

Store signatures

Anyone can contact CA to verify a sig

Sig goes with domain -- "bank.com"

2. e.g. Connect To Bank

Connect to bank.com

Get back signed doc

Verify sig with CA

SSL

Encrypts -> Secrecy

Verifies CA -> Authenticity

Browser CA support

Other Problems

1. Man In The Middle / Spoofing

IP/DNS Spoofing

Solution: digital sigs

2. Replay Attack

Problem

e.g. "ok ship it" message sent from store to warehouse
observed by bad guy

bad guy puts in their address and replays the message to the warehouse.

The signature will be correct!

Solution

1. puts ancillary info like the addr and goods and current time and serial id in the "ship it" message and encrypt and sign the whole thing. The warehouse needs to verify the integrity of each ship it message.

3. "Library" problem

Problem

Person is browsing at the local library. The get up and walk away from the browser in mid session.

Solution

Don't trust info from a form too much.

Ask for the password again at the critical "ship it" step.

4. Denial of service

Send lots of traffic to, say, port 80

Partial Solution

Many attacks rely on ill-formed packets. Upgrade router semantics across the whole internet so only valid packets get through. This cuts down on DOS attacks, but they are still possible.

Other Tech

1. Firewall

Separate your internal net from the rest of the internet. Only allow certain port#'s connections to go through in certain directions. Easier than securing your whole internal network, but can cause problems where some services no longer work since they don't use one of the approved port #'s.

2. Virtual Private Network

Set up an encryption layer in software between all your computers on the Internet. They can talk to each other, but it's 100% encrypted.

Secrecy Ramifications

1. The Costly Security Officer

2. Privacy vs. Transparency

It's a popular notion that people want "privacy", but this is (IMHO) a mistake.

Marketing Harassment

People do not want to be harassed with marketing. Pass laws to achieve **that**, but leave transparency in place.

Privacy

darkness, guns without fingerprints, cars without license plates, paper shredders

Transparency

sunlight, license plates, finger prints, a paper trail -- the truth
"Truth favors the just cause" -- Gandi

193i Lessons

1. Network effects

Determine winners/losers

2. Standards

Bring in participants

Create network effect

Create competition

"2nd best" outcome for vendor

"1st best" outcome rest of us

3. Information Problems

Abundant

Cheaply solved