Reevaluating The Adversary Model for DNS Security

Depending on The Kindness of Strangers

R. Harrison B. Benshoof

Department of Computer Science
Geoorgia State University

2/20/2014

Outline

SSL/TLS

Alice wishes to Start a communication with Bob.
Alice already knows the Certificate Authority's Public key: PK_C

- A$\Rightarrow$ B: "ClientHello" // Initiate exchange
- B$\Rightarrow$ A: PK_B signed by PK_C // Send the certificate
- A$\Rightarrow$ B: $E(PK_B$, NONCE) // Sends a key for the stream cipher

Certificate Authorities

DNSSEC

Key Exchange is Key

The most vulnerable point in secure commnuication is exchanging keys.
With only symetric encyrption, if the adversery sees your key:

- The adversary can read all messages
- The adversary can send false messages

With asymetric encryption, if the adversary can intercept and replace messages from both parties:

- The adversary can read all messages
- The adversary can send false messages
- The adversary can block legitimate message

Asymetric encryption just makes it more difficult to intercept communication, not impossible

Trusted Third Parties solve everything

If both parties have a secure connection to a trusted third party:

- The third party can be used to verify each other's keys
- The third party can be used to detect attempted attacks

CAs breached

2010: VeriSign CA breached

- Kept secret untill 2012
- Full impact not known

2011: DigiNotar CA breached

- Redirected 300,000 Iranian IP addresses using a fraudulent SSL certificate for google.com

2012: Comodo CA breached

- 85,440 forged certificates
- Deemed "To big to fail" and keys were not revoked

2012-Now: Possible systemic NSA interception of SSL traffic

The problems with CAs

- Certificate Authorities get breached
- Certificate Authorities are not inclined to tell us when they are breached. Because they lose money.
- Certificate Authorities are disinclined to revoke compromised keys
- Local goverments have power over Certificate Authortites that secure other contries traffic.

The bottom line: Certificate Authorties are not giving us security

What is a Proof of Work

- Cryptographic Hash functions are designed to make it hard to find two strings which hash to the same value.
- Our best stratagy to find hash collisions is random guessing.
- By allowing for partial matches, we can create a challenge with variable difficulty that is quick to check.
- We can create a string which acts as proof somebody spend time finding a hash collison.

How does Bitcoin Work?

- The bitcoin protocol is centered around maintaing a global state called a "blockchain"
- This state is a list of every transaction ever made."
- Periodically, a new block of signed transactions with a proof of work is added
- The longer this chain becomes, the more difficult it is to falsify

A New SSL Protocol

Alice wishes to Start a communication with Bob.
Alice has setup a CA with Public key: PK_C

- A$\Rightarrow$ CA : $E(PK_{CA}, NONCE_{CA})|E(PK_{CA}, E(NONCE_{CA}, B))$ // Send an encrypted message to the CA with B's info
- CA$\Rightarrow$ B : $E(NONCE_{CA}, PK_B)$ // CA returns B's public key
- A$\Rightarrow$ B : $E(PK_B, NONCE_B)$ //Setup a session key with B

Modifcations to Bitcoin for a key exchange

- Rather than store a list of monitary transactions, we store a list of the ownership and transactions of names
- Limit the length of the blockchain to 1 year
- Add a 'physical' layer check to authenticate new transactions
- Consider alternative incentive methods

Research Directions

- Imropoved SSL
- New DNS distribution Options
- Improved Software Lisensing
- Greater avalibility of PGP style messaging
- Just about anything that needs a key exchange