Reliable client-server connections
Making Telnet secure

Thijs Rozekrans    René Klomp
thijs.rozekrans@os3.nl rene.klomp@os3.nl

System and Network Engineering
University of Amsterdam

July 3, 2013
Introduction

- Authentication of both clients and servers
- Decentralised
- Based on TLS
- Proof of concept
How can current techniques be used to validate the identity of both client and server, using a TLS connection, in a decentralised way?
Motivation

- Increase usage of certificate by clients and servers
- Eliminate the need for certificate authorities
  - Diginotar debacle
  - Foreign governments
  - Centralized
- Techniques are available
- Currently no implementations exist
Design considerations

- PGP or X.509 (CA’s)
- Validating certificates
- Daemon or Library
- Programming language
PGP or X.509

- X.509
  - Widely adapted
  - Validation of certificate is done by CA
- PGP
  - Certificates are managed by users
  - Decentralized design (web-of-trust)
Validating certificates

- Server
  - Server or Client?
    - Server
      - DNS (DANE)
    - Client
      - DNS
      - LDAP
Daemon or Library

- Library
  - Existing GnuTLS library
- Daemon
  - Forwarding mechanism
  - Caching
  - Access to private keys
  - Multiple programming languages
• Performance
• Future extension
Implementation

- Daemon
- Python
- PyGnuTLS Library
- Pass file descriptor of existing connection
Implementation

TCP Handshake

Encrypted

Thijs Rozekrans, René Klomp (UvA)
Reliable client-server connections
July 3, 2013
Implementation

- Based on certificate UID
  - LDAP
  - DANE
- Flags to disable certain checks
- DNSSEC
- Responds with:
  - OK + id
  - ERR + code + message
Implementation

- Forwarding mechanism
- Telnet application as an example
- Possible with every other application
How can current techniques be used to validate the identity of both client and server using a TLS connection in a decentralised way?

- By creating a daemon it is possible!
- Easily implemented using single call to library
- It does work with an existing application (Telnet)
- https://github.com/OS3/rp2_68
Future work

• (D)TLS for UDP and SCTP
• (Soft)HSM
• Caching
• Certificate Pinning
• Libraries in other languages
Are there any questions?

made possible by

[Logos of OpenFortress and nlnet Foundation]