Whoami

Hi everyone
I'm Luca Fulchir,
Italian student at the university of Udine

First time at CCC, nice to meet you all :)

Fenrir

My master thesis project in networks and security

- Transport protocol
- Encryption protocol
- **Authentication protocol**

WARNING: work in progress,
lots of stuff on paper, coding is being done

Why?

- Transport:
  - TCP / UDP
  - SCTP / DCCP
  - Google's QUIC
- Encryption
  - SSL / TLS
  - Google's QUIC
  - (CurveCP, minimaLT...)
- Authentication
  - Kerberos
  - OAuth (unfortunately)

So why a new one?
To use a single library for all your needs, and because, seriously, OAuth?

Transport level

- UDP-based –control flow included
- message-oriented (not only bytestream like TCP, CurveCP)
- multi stream support (SCTP like: multiple messages per pkt)
- short headers (13 bytes + UDP (8) minimum)
- Reliable/unreliable delivery (per-stream)
- ordered/unordered message delivery (per-stream)
- (plus 2 surprises, wait for the last slides)

Packet structure (unencrypted)

bytes	0	1	2	3
0-3	Connection id			
4	padlen	(eventual padding)		
5-8	Stream id		Data length	
9-12	flags	Stream counter		

Connection id → mobile client support, multihoming
green == encrypted
Encrypt-then-MAC

Everything is byte-aligned to avoid inefficiencies

Padding

bytes	0	1	2	3
0-3	Connection id			
4	padlen	(eventual padding)		
5-8	Stream id		Data length	
9-12	flags	Stram counter		

Random padding length: 1 byte per packet, 4 bits per stream
Should help making timing and traffic analisys more difficult
(NEEDS TESTING)

Fenrir also has multiple streams, so CRIME/BREACH gets more difficult

Features

- 4-way handshake with syncookie SCTP style.
    - anonymous connection: 2 RTTs (+ DNSSEC query)
    - authenticated from the beginning: 3 RTTs (+DNSSEC query)
    - federated authentication: 4 RTT → 3 RTT
- stun-like protocol support planned for clients.

DNSSEC

- query for **fenrir**.example.com, type TXT
  fenrir.example.com. 86400 IN TXT “fenrir=”abcdefg....””
- get the public key, udp port, ips... **base85** encoded
- 1300-1400 bytes for a TXT, DNSSEC-signed message

Everything is DNSSEC-signed, so trust is granted.
Easy to change, you have complete control of your zone, multiple servers per ip, single UDP packet DNS response...

Encryption

- perfect forward secrecy
- *NOT* based on SSL/TLS (but similar key exchange)
- ***NOT* based on X.509**: Trust anchor is DNSSEC. (zeroconf for local lan?)
    - *NO PKI mangement!*
    - *NO $$$ per certificate!*

Authentication

- **delayed authentication**: anonymous⇒authenticated in the same connection
- *Federated*: username = user@domain.tld
- ad-hoc federated protocol (passed automated tests, needs more cheking)

3 players (remember kerberos?):

- Authentication Server
- Service
- Client

Authentication

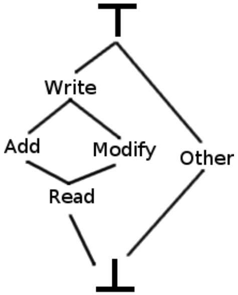
- **token** based (password don't go around too much)
- **NO timestamps** (no clock syncing required!)

An application will run on your device, holding all the account information (tokens).
Your program connects to it (dbus?) and gets keys and connection information, you never see the tokens.

per-device account management (example: phone can access facebook but not bank account, while your pc does both)

Authorization Lattice

- each Service has a Lattice of autorizations:
- If the client has authorization “modify”, it can limit applications to “read”, but not “write”
- have to transfer the lattice :(



Multicast

- NOT yet implemented
- for both reliable and unreliable transmission
- multicast stream + unicast stream for retransmissions

Proxy

HTTPS can not be cached (unless its not so secure anymore).
But Fenrir can!

- protocol-level proxy support!
- transparent / explicit proxy support
- service needs to explicitly tell what to cache, for how long
- resource id needed, something like *example.com/resource/path*
- caching of encrypted material at local or ISP level
- get the decryption key from the real server.

Development status

Just the beginning :(I'm working on the common library
Code not public yet (I want to have something working)

- C++11 standard, as little external libraries as possible
- common library: apache 2.0
- auth daemon: GPLv3
- client: GPLv3 (?)

Help?

- Can't ask too much help on coding – It's my thesis!
- just keep checking; comments & proposals are well accepted

sorry for the long wait, but I have exams, too :(

Thank you!

logo by Piera Zuliani – **piera.zuliani@gmail.com**

Luca Fulchir – **http://fenrirproject.org**

luker@fenrirproject.org