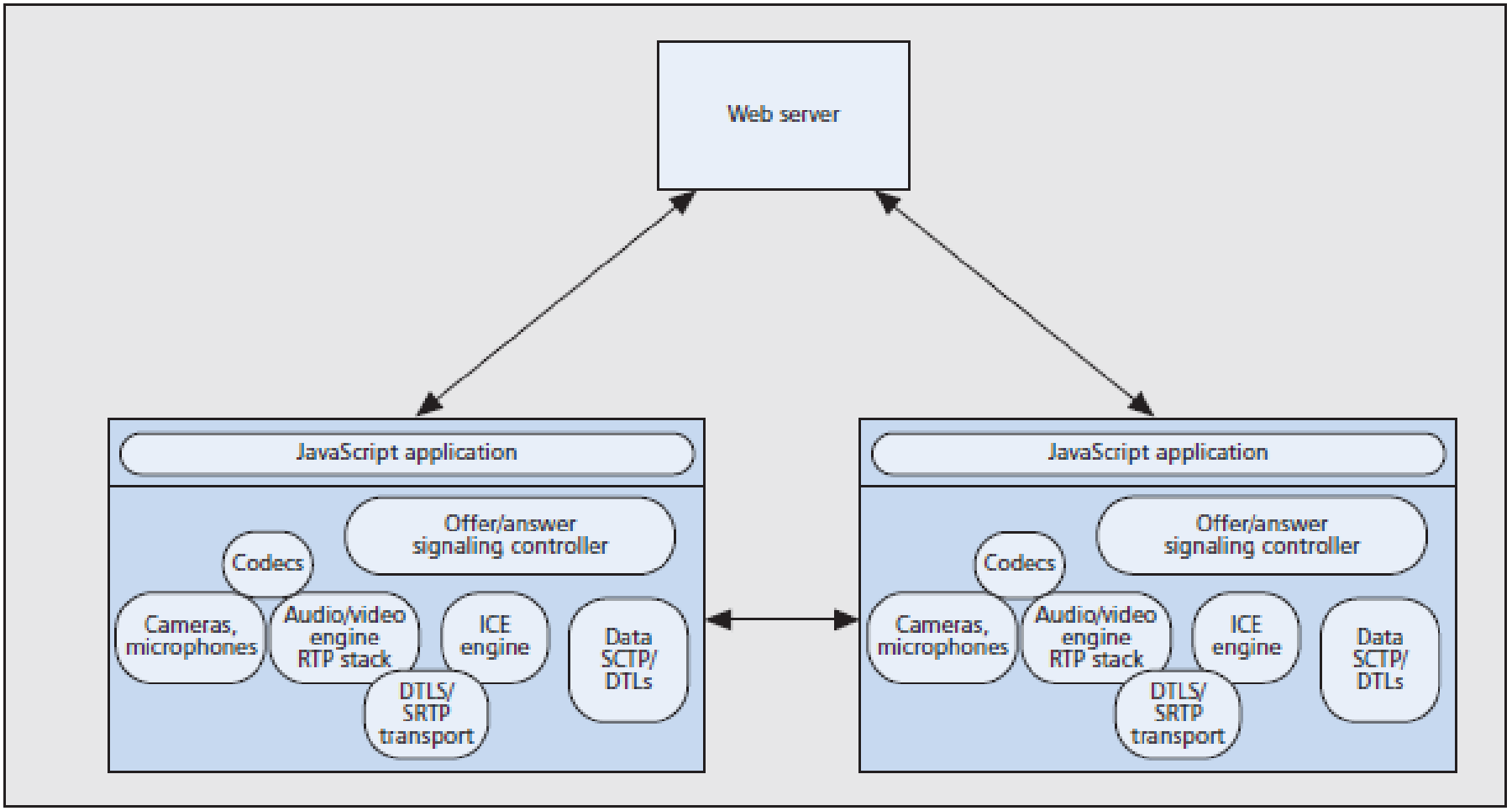
Real-Time Communications for the Web

Presentation of paper by:Cullen Jennings,Ted Hardie,Magnus Westerlund

What is the paper about?

- Describes a peer-to-peer architecture that allows "*direct,interactive,rich communications between browsers and other clients*",called ***WebRTC:***
  - Core components enabled are *peer-to-peer audio and video streams*, and a *multiplexing peer-to-peer data channel*
  - Aims to be an *open standard*,in order to foster innovation
  - Makes use of RTP,SDP,SCTP over DTLS to overcome *transportation and security challenges*
    - (Real-Time Transport Protocol)
    - (Session Description Protocol)
    - (Stream Control Transmission Protocol)
    - (Datagram Transport Layer Security)

Very simply put …



Architecture Description

- (At least) 3 parties participate:
  - An application provider (web server);
  - Two peers (web browsers or other web applications);

How does it happen?

- Each peer downloads a JavaScript application into a local web context (browser or mobile app);

- JavaScript app uses WebRTC API to communicate with local context:
  - peer identifies local media sources as set of "tracks"
  - negotiates connection set-up to other peer
  - passes media along connection …

API Overview

- GetUserMedia
  - returns a MediaStream (local media resources organized into tracks)

- RTCPeerConnection
  - communications channel between peers
  - adding a MediaStream to the channel triggers negotiation with peer
  - no particular protocol or function mandated for peer rendezvous
  - local info is gathered and assembled ,through API, as a SDP offer
  - created data channels use underlying transport built as SCTP over DTLS over UDP
  - A connection involves one pair of peers,but a peer can have multiple connections with the same stream

Security Challenges

- JavaScript can be downloaded from any site without user consent.
- Users must be able to consent to the usage of media resources such as cameras and microphones.
- No significant amount of network traffic must be transmitted without consent from the destination.
- Confidentiality and authentication must be provided for any transmitted media to prevent man-in-the-middle attacks and eavesdropping.
- User private info (identity,geographical location) must not be disclosed when not desired

And how they are handled

- Premise is that web browser or WebRTC implementation functions as trusted computing base
- Together with encrypted and authentificated transport of data;
- Hooks for third-party users can prevent man-in-the-middle attacks;
- Congestion control prevents unfair bandwidth sharing.

- Communication Consent
  - communication for each flow is verified using STUN binding req/resp pairs;
  - use browser-generated secret and 96-bit random transaction identifiers;
  - Continuously verified every 30 s
- Identity
  - “Proxy” in browser – user free to choose identity provider (for ex can be login or cookies)
  - an assertion binds DTLS-SRTP fingerprint with user identity
  - other side checks fingerprint against assertion info

Transportation

- For Real-Time media transport, WebRTC would use Secure RTP profile for RTCP-Based Feedback.
- Key management for SRTP provided by DTLS-SRTP.
- Also a number of standalone extensions …
  - Full Intra Request – joining conferences and switching between media sources;
  - Client-to-mixer audio levels;
  - …

- For arbitrary data transport:
- WebRTC provides reliable and semi-reliable message service, including multiple channels , with prioritization support , between peers.
  - Realized using SCTP with extensions
  - Packets sent over DTLS association, run over lower-layer transport flow,commonly UDP;

Layering maximizes establishment of direct peer-to-peer data channel,using ICE (Interactive Connectivity Establishment)

Transportation Challenges

- NAT and firewall traversal
- Data and Media Multiplexing
- Congestion Control

NAT and Firewall Traversal

- "hole-punching"
  - send STUN packet outside NAT
  - discover IP and port outside NAT
  - works as long as NAT has "endpoint-independent behavior"
  - results in high-quality,cheap connection , but does not work all the time
- media relay box
  - use external media relay box
  - TURN protocol
  - adds latency and jitter to media,but works more often

- ICE protocol finds best possible flow

ICE Protocol

- Try both previous methods, choose best flow
  - both clients assemble a list of possible "flows"
  - candidate flows are tested in order by both sides by sending STUN messages and waiting for a reply
  - best working candidate is chosen

Data and Media Multiplexation

- WebRTC enables multiplexing all RTP media streams and data on a single lower-layer transport flow
  - minimizes risk of NAT and firewall traversal failures
  - fare sharing between data and real-time media

BUT

  - prevents flow-based QoS being applied to sub-flows

However,multiplexation with one flow per RTB session or channel is also supported.

Low-level mechanism most probably UDP.

- Demultiplexation over one single UPD flow
  - done by looking at the value of the first byte of the UDP payload
    - 0 or 1 – STUN packet
    - 20-63 – DTLS packet
    - 128-191 – SRTP/SRTCP packet
    - Implications for using RTP (multiple media types in one RTP session ?)

Congestion Control

- Data channel uses SCTP

- Media channel – is still an open question
  - keep delay and losses low, but provide some kind of fairness

Media Negotiation and Encoding

- Media negotiation using RFC 3264 offer/answer approach
  - basically SDP to exchange media capabilities
  - using SIP
  - SDP negotiates RTP media transport as well as SCTP data channels
  - SDP offer created when adding MediaStreams to the Channel (remember the API ?)
- Media Encoding
  - Negotiation of codecs possible through RTP/SDP , but some codecs mandatory to implement

Conclusions

- WebRTC is an attempt to create an "open eco-system" that facilitates deployment and development of peer-to-peer realtime applications.
- At the moment of writing (April 2013), WebRTC implemented in some browsers
- Right now, WebRTC API is available in:
  - the latest stable version of Chrome
  - the latest stable version of Firefox
  - the latest stable version of Opera
- Tools, resources and tutorials are available at www.webrtc.org

Research Questions

- Real-Time media congestion control?
- (RMCAT Working Group)

- Implications of user consent being used to grand access to local media resources?

- Overall Security Framework Review?