ICE Gathering Errors

Justin Uberti
Peter Thatcher

Problem

When the PeerConnection fails to gather a candidate (because of a bad TURN server, for example), the JS has no idea.

And the JS would like to know when it works, too.

Handling Success: Add URL to ICE Event

```
// Add "url" to onicecandidate. Lets you know which candidates were
// gathered successfully (i.e. from where).
[ Constructor (DOMString type, RTCPeerConnectionIceEventInit eventInitDict)]
interface RTCPeerConnectionIceEvent : Event {
    readonly    attribute RTCIceCandidate candidate;  // exists today
    readonly    attribute DOMString? url;  // the ICE server |candidate| is from
};
```

Handling Failure: Solution A (Generic)

```
// Finally add this.
partial interface RTCPeerConnection {
  attribute EventHandler onerror;  //
}

// With a base class for all errors.
enum RTCPeerConnectionErrorType { "ice-gather" };
interface RTCPeerConnectionErrorEvent : Event {
  readonly attribute RTCPeerConnectionErrorType type;
  readonly attribute DOMString description;
};

// .type == "ice-gather"
interface IceCandidateErrorEvent : RTCPeerConnectionErrorEvent {
  DOMString url;  // RTCIceServer.uri
  int reason;     // 4xx error from TURN server
                  // 6xx indicating general/connection failure
};
```

Handling Failure: Solution B (Specific)

```
// Add this instead.
partial interface RTCPeerConnection {
  attribute EventHandler onicecandidateerror;
}

// Details about the error.
interface IceCandidateErrorEvent : Event {
  DOMString url;  // RTCIceServer.url
  int reason;      // 4xx error from TURN server
                  // 6xx indicating general/connection failure
  DOMString description;  // e.g. "Authorization failed"
};
```