

End-User Services with OpenSIPS

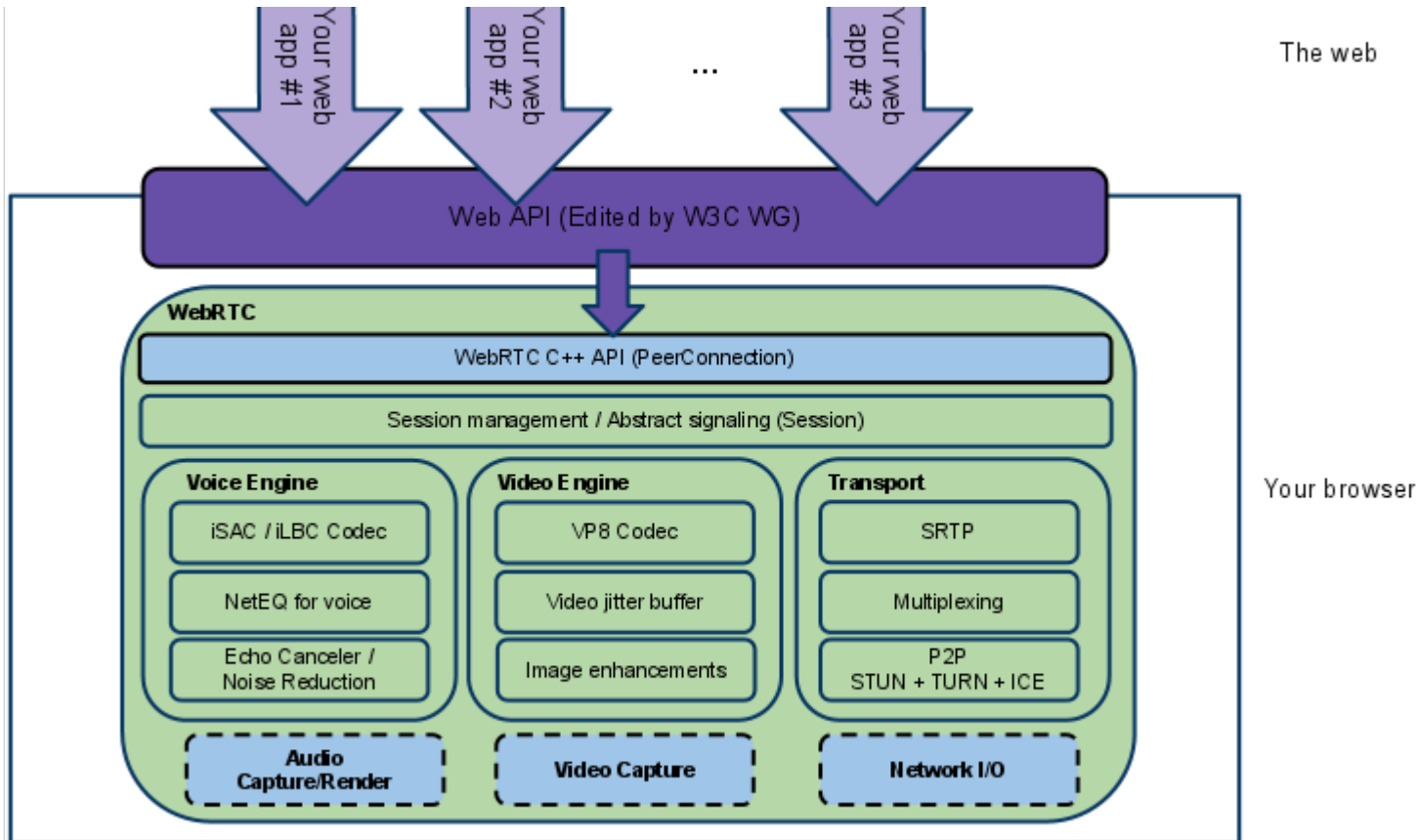
*Vlad Paiu
OpenSIPS Project Developer
OpenSIPS Solutions*

Integrating end-user services into OpenSIPS-based platforms

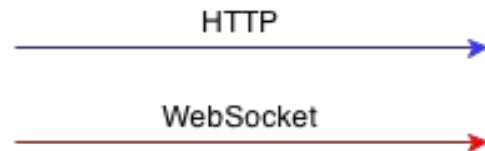
- **Integrating WEBRTC**
- **Integrating MSRP**
- **Integrating Push Notifications**

WEBRTC

Project that enables web browsers to offer Real-Time Communications



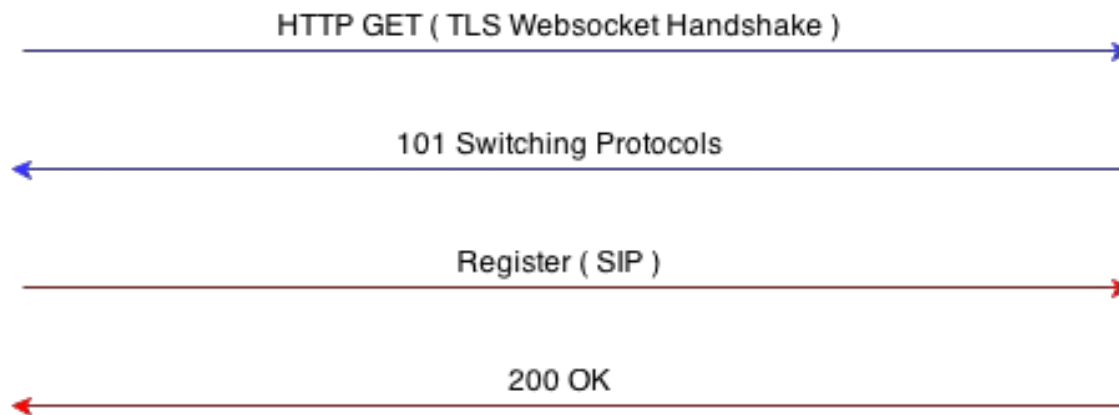
- **WebRTC does not mandate any signaling protocol**
 - SIP over Websockets was standardized in RFC 7118 (January 2014)
- **WebRTC explicitly forbids the use of plain RTP**
 - SRTP is mandatory
- **Mandatory To Implement Voice Codecs are G711 and Opus**

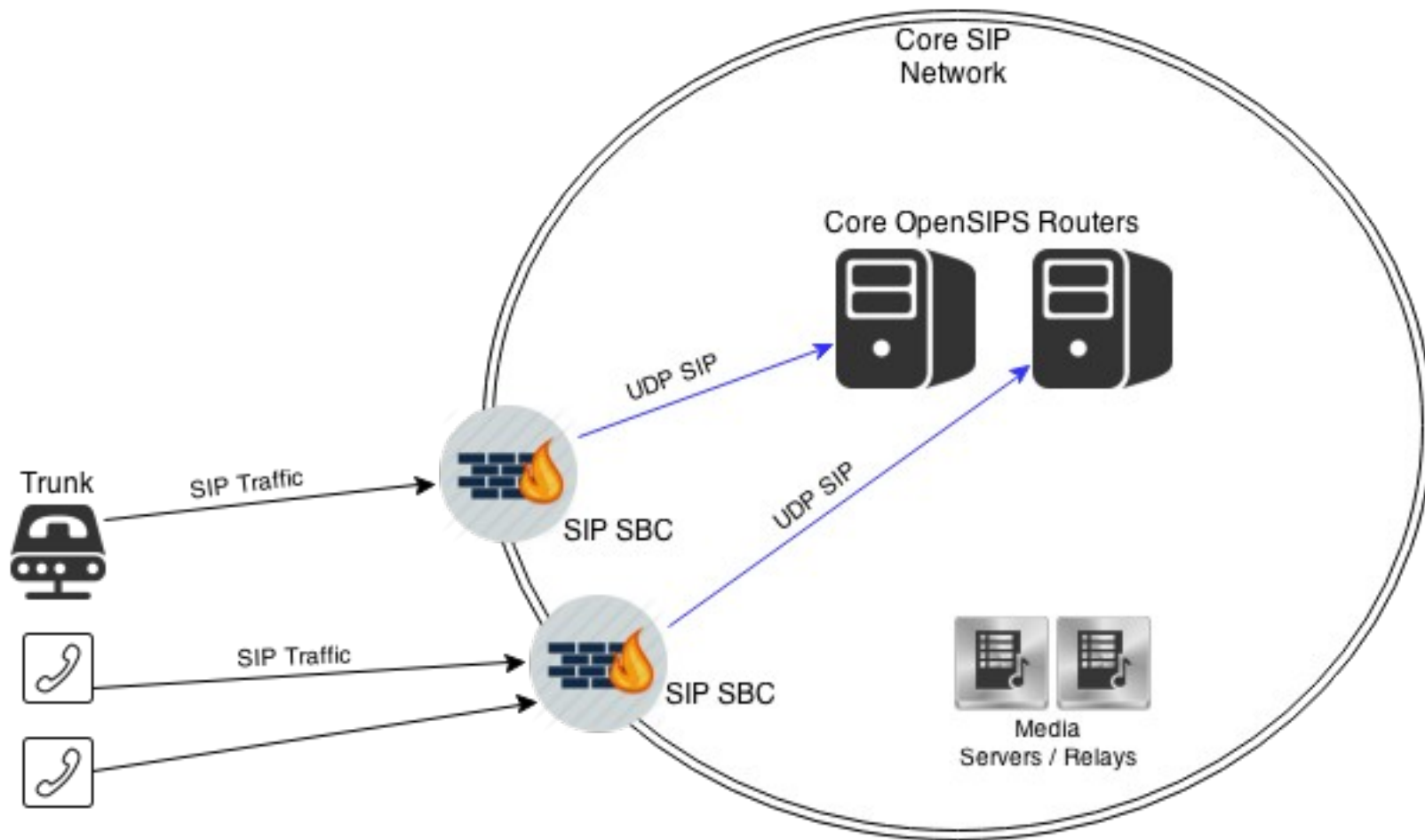


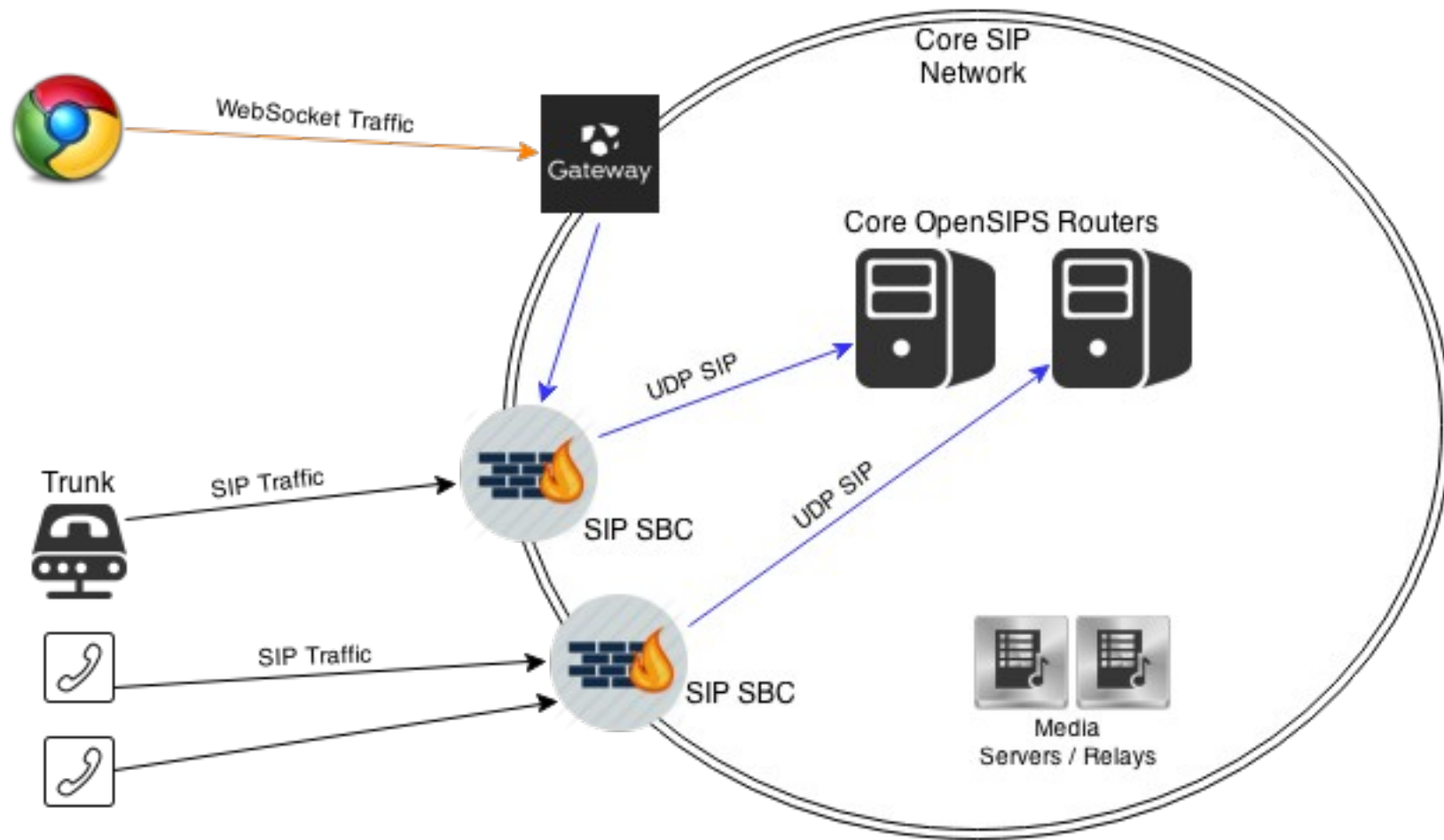
WebRTC Client

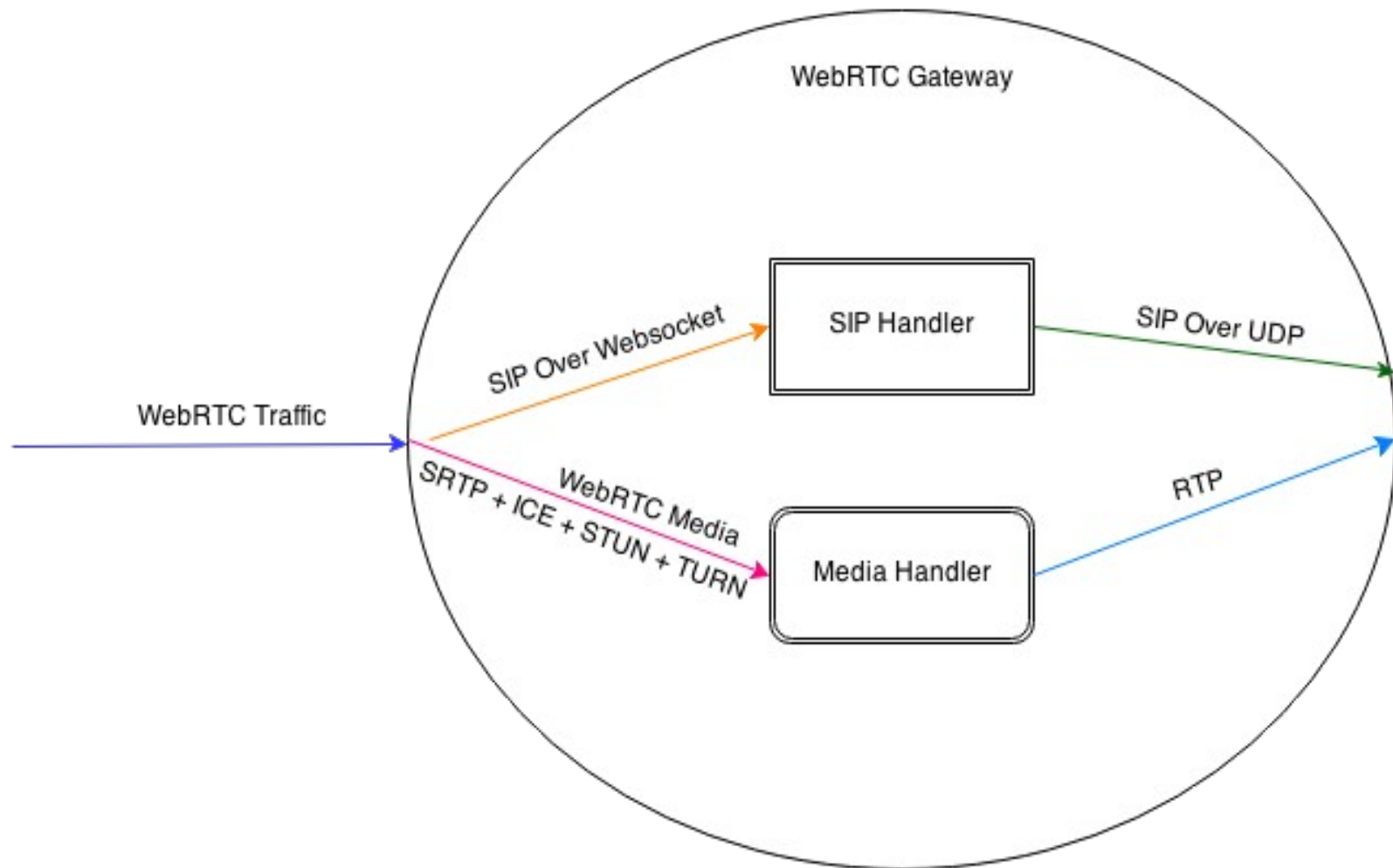


SIP WebSocket Server







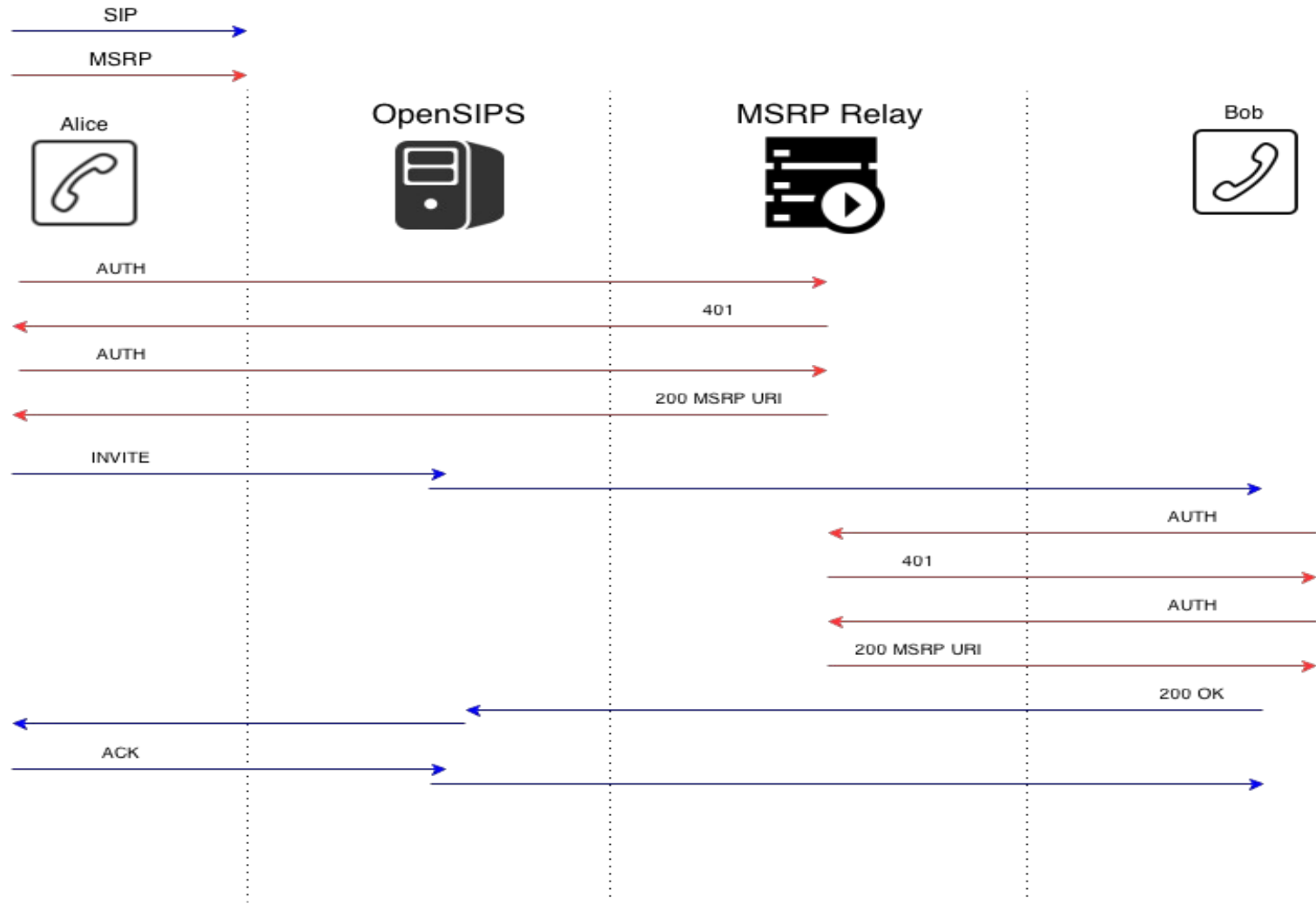


- **Not possible to simply view the Websocket channel as a tunnel**
 - Proxies need to deal differently with messages when Websocket is used, when compared with routing via UDP / TCP
- **Complexity and Performance**
 - SIP Proxies also have to support HTTP
 - Gateway approach is simpler and does not affect existing deployments
- **Security**
 - HTTP being a commonly exploited protocol, It would be unsafe to push it all the way to the core network

- **Websocket to SIP Gateway**
 - OverSIP
- **WEBRTC to SIP**
 - Webrtc2sip
 - Websocket to SIP gateway
 - RTCWeb Breaker (ICE and SRTP to regular SDP & RTP)
 - Media Coder/Decoder
- **Clients**
 - SIPML5
 - JSSIP

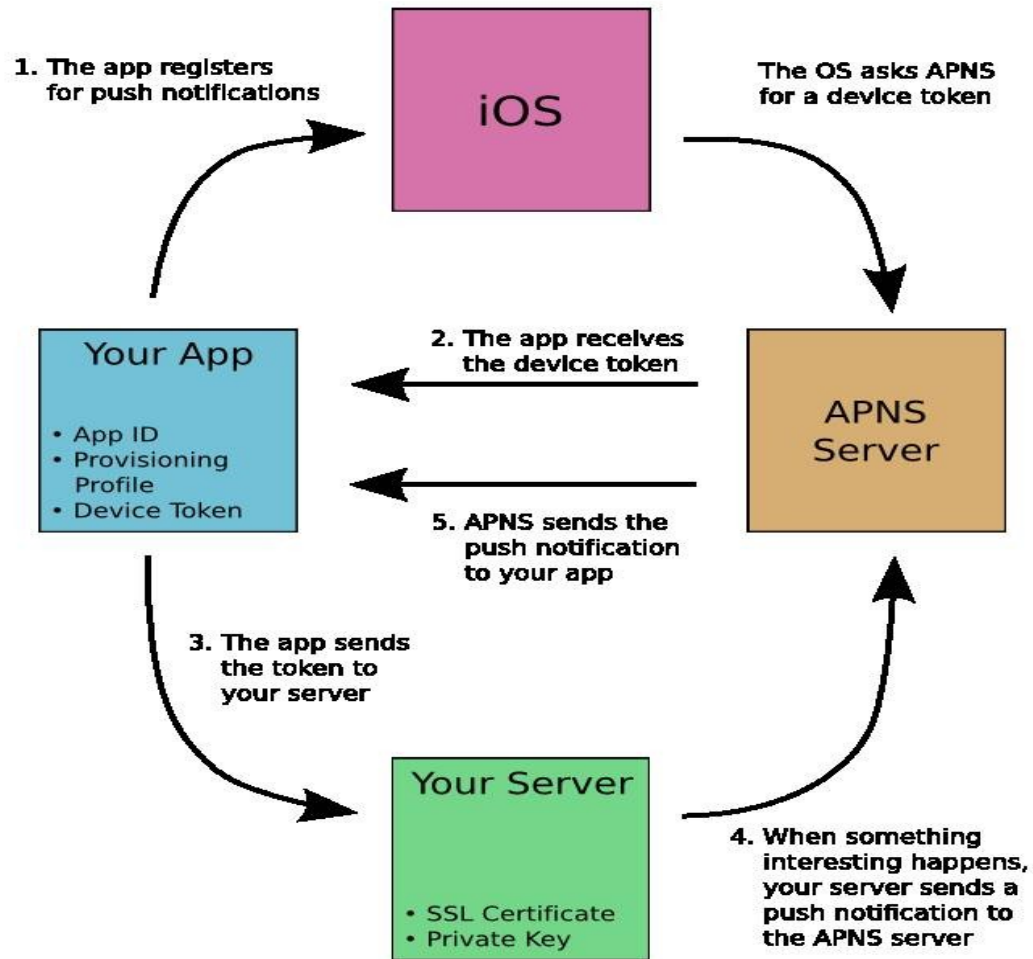
MSRP

- **Message Session Relay Protocol**
- **Generic protocol used for transmitting a series of related messages within a communication session**
 - IM
 - File Transfer
 - Screen sharing
- **MSRP is NOT SIP**
 - Has a similar syntax to SIP
 - A MSRP session is setup through the SIP offer-answer model
 - MSRP not defined for UDP



- **MSRP and SIP routers can and should be separate entities**
- **The MSRP relay can be configured to use the OpenSIPS database for authenticating subscribers**
- **No change to the SIP infrastructure required**
- **MSRPRelay.org**

Push Notifications

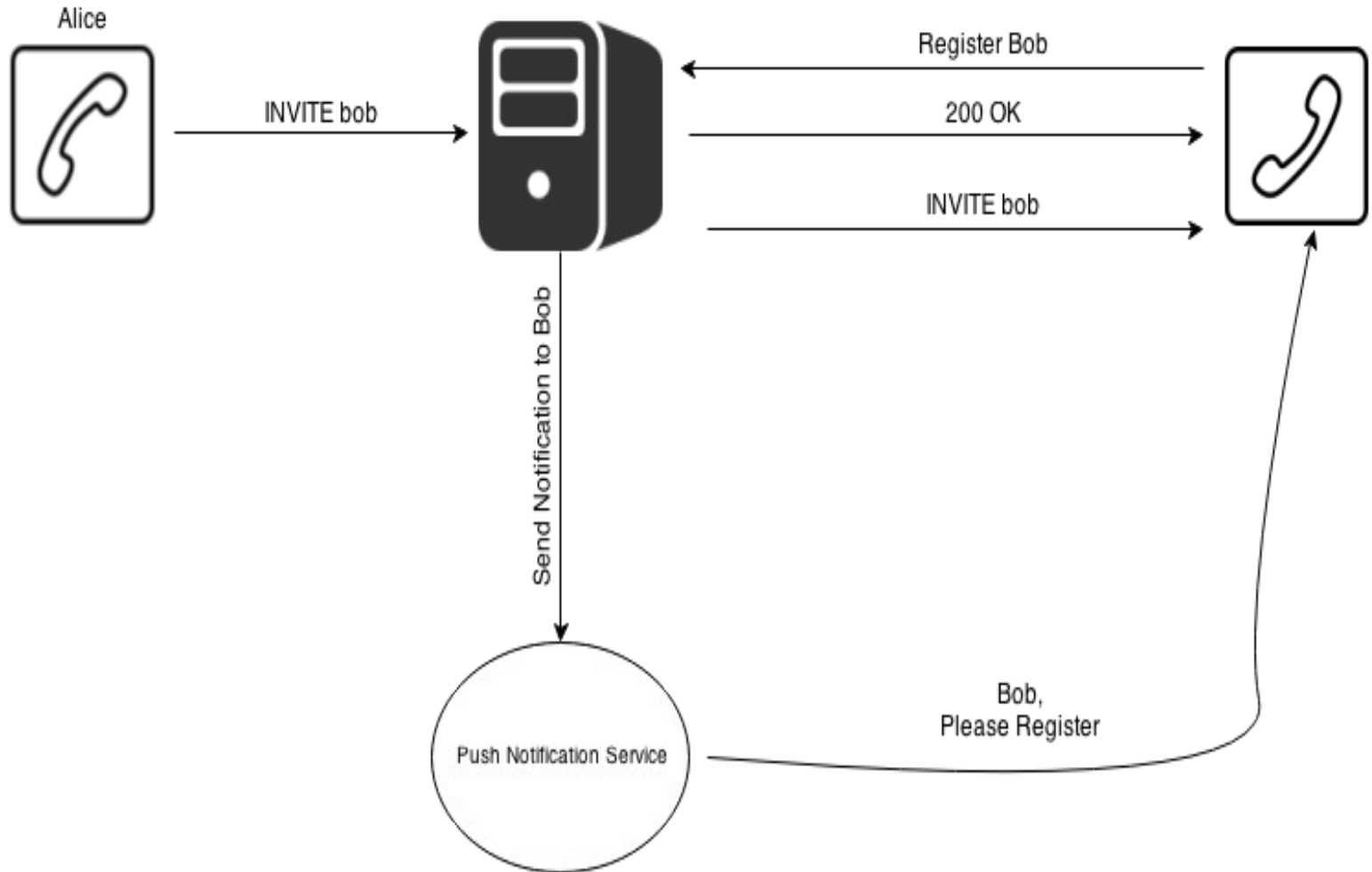


The Good

- **Helps save user's phone battery**
 - Longer battery life => happier user
- **Better interaction with the user for delivering various events**

The Bad

- **Not reliable / Not guaranteed to arrive at the device**
- **Can take a long time to complete**
- **More load on the SIP server (if not implemented right)**



Demo

```
route{
...
  if (!lookup("location","m")) {
    # send 180 so caller hears ringing while we do the push notification
    send_reply("180","Ringing");
    # call our push notification script
    exec_msg("/push_notification.sh $rU $fU");
    # send the request to /dev/null equivalent ( port 9 )
    $du = "sip:127.0.0.1:9";
    # wait a max of 2 seconds for the push notification to work
    $T_fr_timeout = 2;
    t_on_failure("after_push_notification");
    route(relay);
  }
}
failure_route[after_push_notification] {
...
  $du = NULL;
  if (!lookup("location")) {
    # still not registered :(
    t_reply("404","Not Found");
    exit;
  }
  route(relay);
}
```

- **OpenSIPS can successfully help in providing various end-user services**
- **Use the proper tool for the proper job**

Thank you for your attention

**You can find out more at www.opensips.org
vladpau@opensips.org**

Questions are welcome