



Mongoose

Citizen of the World (Wide Web)

Who are you?

- Webmaster of social site?
- Designer of corporate page?
- Backend developer?
- Geek with full stomach after lunch?

How can you embed a chat?

- Combine PHP, JavaScript and storing messages in MySQL.
- Use existing plugin/library and external chat server.
- Make custom server with REST/Websockets interface.
- Or...



Host your own chat

- All you need is dedicated server where you can install Erlang, MongooseIM and open chosen ports in firewall.
- May use the server with webpage.
- Integrate existing JavaScript XMPP libraries with MongooseIM BOSH and Websockets endpoints.

Why XMPP?

- Proven, popular protocol
- Lots of extensions
- Many server and client implementations
- Easy to comprehend:

```
<message from="john@example.com/macbook"  
to="alice@example.com/nokia">  
  <body>Hello Alice!</body>  
</message>
```


XEP-0239: Binary XMPP

- Traditional stanza: <presence/>

- Binary representation:

```
00111100011100000111001001100101011100110110  
01010110111001100011011001010010111100111110
```

- Binary XMPP representation:

```
<zero/><zero/><one/><one/><one/><one/><zero/><zero/>  
<zero/><one/><one/><one/><zero/><zero/><zero/><zero/>  
<zero/><one/><one/><one/><zero/><zero/><one/><zero/>  
<zero/><one/><one/><zero/><zero/><one/><zero/><one/>  
<zero/><one/><one/><one/><zero/><zero/><one/><one/>  
<zero/><one/><one/><zero/><zero/><one/><zero/><one/>  
<zero/><one/><one/><zero/><one/><one/><one/><zero/>  
<zero/><one/><one/><zero/><zero/><zero/><one/><one/>  
<zero/><one/><one/><zero/><zero/><one/><zero/><one/>  
<zero/><zero/><one/><zero/><one/><one/><one/><one/>  
<zero/><zero/><one/><one/><one/><one/><one/><zero/>
```


Our little carnivore

- Very strong claws to dig through lots of messages
- Able to deal with multiple ~~meals~~ servers in cluster
- Adopts very well to ~~Eurasian and African~~ modern Web environment.



Getting Mongoose

- Download package from ESL website or repo
- Clone git repository:
<https://github.com/esl/ejabberd>

Mongoose's reflex

Server:

- AMD FX-8150 8x3.3GHz
- 32GB RAM

One big room:

- 3000 users
- 10 messages per second

Multiple rooms:

- 2000 rooms
- 25 users/room
- 0.6 message per second per room

Case Study: Erlang Central

Erlang Factory streams

- Groupchats for live streams were first.
- Use MUCk1 (with JSJaC as backend) embedded in a very straightforward way (`<iframe>`).
- Taps into Wordpress cookies and uses them for authenticating users.

Client-side

- Connects to MongooseIM via Websockets
- Authenticates with user login and Wordpress 'logged in cookie'
- Cookie format:
`piotr_nosek|1368010779|49aa66843380c377e93b198b966eb699`
`user login exp. date magic hash`

Steps to create magic hash

1. `Frag = password_hash[8:11]`
2. `HMACKey = logged_in_key + logged_in_salt`
3. `HMACData = username + Frag + '|' + ExpTime`
4. `HMAC = hmac(md5, HMACData, HMACKey)`
5. `MagicHash = hmac(md5, username + '|' + ExpTime, HMAC)`

Server-side module

`ejabberd_auth_wordpress.erl`

- Reproduces Wordpress 'logged in cookie'
- Compares it with provided password (cookie)
- Generated from user password hash fragment, user login, expiration date and salt defined by administrator

Case Study: Erlang Central

Erlang Central Cafe

- Facebook-like global groupchat
- Uses xmpptk (with JSJaC backend, like stream chats)
- Chosen BOSH instead of Websockets
 - Websockets are closed on page reload
 - Test drive for MongooseIM reimplemented BOSH support

Client-side

- Uses cookie generated in similar way to 'logged in cookie'.
- Modified xmpptk and jQuery chat box to fit into Wordpress plugin and match Erlang Central look and feel.

Server-side

- Compares password (cookie) against generated hash.
- New BOSH implementation.
- For 3rd party clients: hashes password with Erlang implementation of phpass and checks against hash stored in DB.

(Side) Effects of Erlang Central chats

- Real application for Websockets and new BOSH support
- People watching stream have a way of telling camera guy to show slides, not just speaker face. :)
- Modules for integrating Erlang with Wordpress and/or PHP:
 - Cafe plugin
 - phpass implementation

Useful links

- MongooseIM load tests - full article
<http://soerlang.tumblr.com/post/35563581435/esl-mongooseim-vs-p1-ejabberd-part-i-piotr-nosek>
- MUCk1
<https://github.com/sstrigler/MUCk1>
- xmpptk
<https://github.com/sstrigler/xmpptk>
- Wordpress auth for MongooseIM:
<https://github.com/esl/ejabberd>, 'wordpress' branch
- Cafe plugin: To Be Released
- Detailed tutorials (self-advertisement ;))
<http://getcookie.wordpress.com>

Thank you!

Köszönöm!



And see you again in `erlang:get_cookie()` blog. ;)