



# FHQ 2014 Manual's

Doc-Version: 13

Doc-Date-Build: January 14, 2015

Pages: 18

Dev.: Evgenii Sopov

Dev.: Dmitrii Mukovkin

Tomsk

Russian Federation

Web-site: <http://fhq.keva.su>

E-mails: [mrseakg@gmail.com](mailto:mrseakg@gmail.com)

With support by



# Contents

|          |                                      |          |
|----------|--------------------------------------|----------|
| <b>1</b> | <b>Configuration's</b>               | <b>2</b> |
| 1.1      | MySQL (or MariaDB)                   | 2        |
| 1.2      | iptables                             | 2        |
| 1.3      | Firewall on CentOS 7 (use firewalld) | 3        |
| 1.4      | Apache                               | 3        |
| 1.5      | PHP5                                 | 4        |
| <b>2</b> | <b>API</b>                           | <b>5</b> |
| 2.1      | Auth                                 | 6        |
| 2.1.1    | Sign in/Login                        | 6        |
| 2.1.2    | Restore Password                     | 6        |
| 2.1.3    | Registration                         | 7        |
| 2.2      | Games                                | 8        |
| 2.2.1    | List of games                        | 8        |
| 2.2.2    | Get Game Info                        | 9        |
| 2.2.3    | Select Game                          | 10       |
| 2.2.4    | Scoreboard                           | 11       |
| 2.2.5    | Refresh User's Score                 | 11       |
| 2.2.6    | Insert Game Info                     | 12       |
| 2.2.7    | Update Game Info                     | 13       |
| 2.2.8    | Remove Game                          | 14       |
| 2.3      | Users                                | 15       |
| 2.3.1    | Get List Users                       | 15       |
| 2.3.2    | Get User Info                        | 15       |
| 2.3.3    | Get IP-Addresses of User             | 15       |
| 2.3.4    | Insert User Info                     | 16       |
| 2.3.5    | Update User Info                     | 16       |
| 2.3.6    | Insert User Info                     | 17       |
| 2.3.7    | Delete User                          | 17       |
| 2.3.8    | Update User Location                 | 17       |
| 2.3.9    | Update User Style                    | 18       |

# Chapter 1

## Configuration's

### 1.1 MySQL (or MariaDB)

Install on server git-core.

```
root@debian:~# apt-get install git-core
```

Clone repository:

```
root@debian:~# git clone https://github.com/sea-kg/fhq fhq.github.temp
root@debian:~# mysql -u root -p -h localhost \
< fhq.github.temp/sql/init_database_freehackquest.sql
```

or look sql-script

```
https://github.com/sea-kg/fhq/blob/master/sql/init\_database\_freehackquest.sql
```

### 1.2 iptables

Some settings.

Linux and iptables.

Add to the firewall permission for the server (if it is not on the local machine)

```
# iptables -A INPUT -p tcp -s 0/0 --sport 1024:65535 -d 172.16.53.102 \
--dport 3306 -m state --state NEW,ESTABLISHED -j ACCEPT
# iptables -A OUTPUT -p tcp -s 172.16.53.102 --sport 3306 -d 0/0 \
--dport 1024:65535 -m state --state ESTABLISHED -j ACCEPT
# iptables-save
```

Check that the records were

```
# iptables -L INPUT -n -v --line-numbers
# iptables -L OUTPUT -n -v --line-numbers
```

To delete an record: 1. find a number

```
# iptables -L INPUT -n -v --line-numbers
```

2. and delete

```
# iptables -D INPUT number_of_your_entry
```

And of course, keep

```
# iptables-save
```

## 1.3 Firewall on CentOS 7 (use firewalld)

Some settings.

CentOS and MariaDB and Firewall.

CentOS 7 + firewalld:

```
# firewall-cmd --zone=public --add-port=3306/tcp --permanent
# firewall-cmd --reload
```

## 1.4 Apache

Setting up a domain name.

Add to /etc/apache2/sites-available/default next text:

```
<VirtualHost *:80>
    Options -Indexes FollowSymLinks MultiViews
    DocumentRoot /var/www/fhq/
    ServerName fhq.keva.su
    ErrorLog /var/log/apache2/fhq.keva.su-error_log
    CustomLog /var/log/apache2/fhq.keva.su-access_log common

    <Directory "/var/www/fhq/files">
        AllowOverride None
        Options -Indexes
        Order allow,deny
        Allow from all
    </Directory>

    <Directory /var/www/fhq/config>
        Order deny,allow
        Deny from all
    </Directory>
</VirtualHost>
```

OR you can copy the file from

```
https://github.com/sea-kg/fhq/blob/master/manual/fhq.config.example
```

to file `"/etc/apache2/sites-available/fhq.config"`

## 1.5 PHP5

Install and configure

```
root@debian# apt-get install php5
```

If not working captcha:

```
root@debian# apt-get install php5-gd
```

To work with sending e-mails:

```
root@debian# apt-get install php-pear
root@debian# pear install Mail-1.2.0
root@debian# pear install Net_SMTp
```

Install git:

```
root@debian# apt-get install git-core
```

Download source code:

```
root@debian# cd ~
root@debian~# git clone https://github.com/sea-kg/fhq.git fhq.git
root@debian~# cd fhq.git
```

Then just do a link:

```
root@debian~# ln -s "'pwd'/php/fhq" "/var/www/fhq"
```

Next, copy and configure:

```
root@debian~# cp /var/www/fhq/config.php.inc /var/www/fhq/config.php
```

# Chapter 2

## API

This chapter will talk about what external functions exist to work with the system. Also presented are various examples for the job. Also, this chapter is devoted to the frontend developers using kernel of fhq.

## 2.1 Auth

### 2.1.1 Sign in/Login

This function access for all.

URL:

```
http://fhq.keva.su/api/auth/sign_in.php
```

Input parameters (GET or POST):

- **email** - identifier of user
- **password** - password of user
- **client** - identifier for client (fhq-web or other frontends)

Successfully response:

```
{
  "result": "ok",
  "token" : "76558894-0AA9-11E4-09F0-D353D3CF86D5"
}
```

Code errors (10\*):

- **Error 1001** - parameters was not found email or password
- **Error 1002** - email or password was not found in system

### 2.1.2 Restore Password

This function access for all.

URL:

```
http://fhq.keva.su/api/auth/restore.php
```

Captcha URL:

```
http://fhq.keva.su/captcha.php
```

Input parameters (GET or POST):

- **email** - identifier of user
- **captcha** - captcha

Successfully response:

```
{
  "result": "ok",
  "data": {
    "message": "Check your your e-mail (also check spam)."
```

Also look ??

### 2.1.3 Registration

This function access for all.

URL:

```
http://fhq.keva.su/api/auth/registration.php
```

Captcha URL:

```
http://fhq.keva.su/captcha.php
```

Input parameters (GET or POST):

- **email** - new identificator for user (email)
- **captcha** - captcha

Successfully response:

```
{
  "result": "ok",
  "data": {
    "message": "Check your your e-mail (also check spam)."
```

Also look ??



## 2.2 Games

### 2.2.1 List of games

This function access for all authorized users.

URL:

```
http://fhq.keva.su/api/games/list.php
```

// TODO paging Input parameters (GET or POST):

- no

Successfully response:

```
{
  "result": "ok",
  "data": {
    "2014-12-20 23:30:00": {
      "id": "5",
      "title": "FHQ 2014",
      "type_game": "jeopardy",
      "date_start": "2014-12-20 23:30:00",
      "date_stop": "2014-12-31 23:00:00",
      "date_restart": "2015-01-02 20:00:00",
      "description": "The game will be last<br>\nFor searching hackers<br>\n<a href='\"http://cs7009.vk.me/c625821/v625821429/ff45/dYYxQRlVhJg.jpg\"'>http://cs7009.vk.me/c625821/v625821429/ff45/dYYxQRlVhJg.jpg",
      "logo": "http://cs7009.vk.me/c625821/v625821429/ff45/dYYxQRlVhJg.jpg",
      "owner": "220",
      "nick": "[keva] sea-kg",
      "permissions": {
        "delete": true,
        "update": true
      }
    },
    "2014-04-01 10:00:00": {
      "id": "4",
      "title": "SibirCTF 2014 J (Copy)",
      "type_game": "jeopardy",
      "date_start": "2014-04-01 10:00:00",
      "date_stop": "2014-04-01 15:00:00",
      "date_restart": "2014-04-02 09:00:00",
      "description": "Copies tasks from\n<a href='\"http://sibirctf.org\"'>sibi",
      "logo": "http://fhq.keva.su/files/icons/logo_sibirctf2014.png",
      "owner": "220",
      "nick": "[keva] sea-kg",
      "permissions": {
        "delete": true,
        "update": true
      }
    }
  }
}
```

```

    }
  },
  "2013-10-06 09:00:00": {
    "id": "3",
    "title": "FHQ 2013",
    "type_game": "jeopardy",
    "date_start": "2013-10-06 09:00:00",
    "date_stop": "2013-10-18 21:00:00",
    "date_restart": "2013-10-20 09:00:00",
    "description": "Game for search hackers",
    "logo": "files\\orig\\minilogo_old.png",
    "owner": "220",
    "nick": "[keva] sea-kg",
    "permissions": {
      "delete": true,
      "update": true
    }
  },
  "2012-10-06 09:00:00": {
    "id": "1",
    "title": "FHQ 2012",
    "type_game": "jeopardy",
    "date_start": "2012-10-06 09:00:00",
    "date_stop": "2012-10-17 16:00:00",
    "date_restart": "2012-10-19 09:00:00",
    "description": "The game for search hackers",
    "logo": "files\\orig\\minilogo_old.png",
    "owner": "220",
    "nick": "[keva] sea-kg",
    "permissions": {
      "delete": true,
      "update": true
    }
  }
},
"current_game": "5",
"permissions": {
  "insert": true
}
}

```

Also look ??

## 2.2.2 Get Game Info

This function access for all.

URL:

```
http://fhq.keva.su/api/games/get.php
```

Input parameters (GET or POST):

- **id** - identifier of game (integer)

Successfully response:

```
{
  "result": "ok",
  "data": {
    "id": "5",
    "type_game": "jeopardy",
    "title": "SibirCTF 2014 J",
    "date_start": "2014-03-27 00:00:00",
    "date_stop": "2014-03-28 00:00:00",
    "date_restart": "2014-04-01 08:00:00",
    "description": "Game was offline at 1 April 2014 in Tomsk",
    "logo": "logo_2014-03-14_starosta.png",
    "owner": "46"
  }
}
```

??

### 2.2.3 Select Game

This function access for all.

URL:

```
http://fhq.keva.su/api/games/choose.php
```

Input parameters (GET or POST):

- **id** - identifier of game (integer)

Successfully response:

```
{
  "result": "ok",
  "data": {
    "id": "5",
    "type_game": "jeopardy",
    "title": "SibirCTF 2014 J",
    "date_start": "2014-03-27 00:00:00",
    "date_stop": "2014-03-28 00:00:00",
```

```
        "date_restart": "2014-04-01 08:00:00",
        "logo": "logo_2014-03-14_starosta.png",
        "owner": "46",
        "description": "Game was offline at 1 April 2014 in Tomsk",
    }
}
```

Also look ??

## 2.2.4 Scoreboard

This function access for all.

URL:

```
http://fhq.keva.su/api/games/scoreboard.php
```

// TODO paging Input parameters (GET or POST):

- **gameid** - identifier of game (integer)

Successfully response:

```
{
  "result": "ok",
  "data": {
    "1": [{
      "nick": "sea-kg",
      "score": "1161"
    }],
    "2": [{
      "nick": "user",
      "score": "111"
    }, {
      "nick": "user1",
      "score": "111"
    }]
  },
  "gameid": "7"
}
```

Also look ??

## 2.2.5 Refresh User's Score

This function access for all authorized users.

Must be selected game.

Refreshing user's score.

URL:

```
http://fhq.keva.su/api/games/update_score.php
```

Input parameters (GET or POST):

- no

Successfully response:

```
{
  "result": "ok",
  "data": [],
  "user": {
    "score": "1161"
  }
}
```

Also look ??

## 2.2.6 Insert Game Info

This function access only for admin.

URL:

```
http://fhq.keva.su/api/games/insert.php
```

Input parameters (GET or POST):

- **uuid\_game** - global identificator of the game (text)
- **title** - name of the game (text)
- **logo** - link to image (text)
- **type\_game** - type of the game (text) possible values: "jeopardy" or "attack-defence"
- **date\_start** - start date of the game (datetime, format: '0000-00-00 00:00:00')
- **date\_stop** - stop date of the game (datetime, format: '0000-00-00 00:00:00')
- **date\_restart** - restart date of the game (datetime, format: '0000-00-00 00:00:00')
- **description** - description of the game

Successfully response:

```
{
  "result": "ok",
  "data" : {
    "game" : {
      "id" : 2
    }
  }
}
```

Also look ??

### 2.2.7 Update Game Info

This function access only for admin.

URL:

```
http://fhq.keva.su/api/games/update.php
```

Input parameters (GET or POST):

- **id** - identifier of the game (number)
- **title** - name of the game (text)
- **logo** - link to image (text)
- **type\_game** - type of the game (text) possible values: "jeopardy" or "attack-defence"
- **date\_start** - start date of the game (datetime, format: '0000-00-00 00:00:00')
- **date\_stop** - stop date of the game (datetime, format: '0000-00-00 00:00:00')
- **date\_restart** - restart date of the game (datetime, format: '0000-00-00 00:00:00')
- **description** - description of the game

Successfully response:

```
{
  "result": "ok"
}
```

Also look ??

### 2.2.8 Remove Game

This function access only for admin.

URL:

```
http://fhq.keva.su/api/games/delete.php
```

Captcha URL:

```
http://fhq.keva.su/captcha.php
```

Input parameters (GET or POST):

- **id** - identificator of game (integer)
- **captcha** - captcha (text)

Successfully response:

```
{  
  "result": "ok"  
}
```

Also look ??

## 2.3 Users

### 2.3.1 Get List Users

This function access for admin.

URL:

```
http://fhq.keva.su/api/users/list.php?userid=46
```

Input parameters (GET or POST):

- **userid** - identifier of user (integer)

Successfully response:

```
{  
  // TODO  
}
```

??

### 2.3.2 Get User Info

This function access for authorized users.

URL:

```
http://fhq.keva.su/api/users/get.php?userid=46
```

Input parameters (GET or POST):

- **userid** - identifier of user (integer)

Successfully response:

```
{  
  // TODO  
}
```

??

### 2.3.3 Get IP-Addresses of User

This function access for admin.

URL:

```
http://fhq.keva.su/api/users/get_ips.php?userid=46
```

Input parameters (GET or POST):



- **userid** - identifier of user (integer)

Successfully response:

```
{  
  // TODO  
}
```

??

### 2.3.4 Insert User Info

This function access for admin.

URL:

```
http://fhq.keva.su/api/users/insert.php?userid=46&todo
```

Input parameters (GET or POST):

- **TODO**

Successfully response:

```
{  
  // TODO  
}
```

??

### 2.3.5 Update User Info

This function access for authorized users.

TODO

URL:

```
http://fhq.keva.su/api/users/get.php?userid=46
```

Input parameters (GET or POST):

- **userid** - identifier of user (integer)

Successfully response:

```
{  
  // TODO  
}
```

??

### 2.3.6 Insert User Info

This function access for admin.

URL:

```
http://fhq.keva.su/api/users/insert.php?userid=46&todo
```

Input parameters (GET or POST):

- **TODO**

Successfully response:

```
{  
  // TODO  
}
```

??

### 2.3.7 Delete User

This function access for admin.

URL:

```
http://fhq.keva.su/api/users/delete.php?userid=46
```

Input parameters (GET or POST):

- **userid** - identifier of user (integer)

Successfully response:

```
{  
  // TODO  
}
```

??

### 2.3.8 Update User Location

Only for authorized user.

URL:

```
http://fhq.keva.su/api/users/get.php?userid=46
```

Input parameters (GET or POST):

- **country** - identifier of user (integer)

- **city** - identifier of user (integer)
- **university** - identifier of user (integer)

Successfully response:

```
{
  // TODO
}
```

Code Errors:

- **909** - Not found parameter "country"
- **910** - Not found parameter "city"
- **915** - Not found parameter "university"
- **911** - Problem with database or query

### 2.3.9 Update User Style

This function access for authorized users.

URL:

```
http://fhq.keva.su/api/users/get.php?userid=46
```

Input parameters (GET or POST):

- **userid** - identifier of user (integer)

Successfully response:

```
{
  // TODO
}
```

??