Comunicacions

La(es) aplicacions es comuniquen amb el servidor mitjançant requeriments http(s).

Segons lo suggerit per J.Morell farem servir una única URL per totes les operacions, s’accedirà sempre mitjançant un GET i les comunicacions es faran per https.

Els paràmetres visibles es diran “op” per indicar la operació que volem fer e “id” per identificar el participant. En principi poden der numèriques o alfanumèriques segons ens convingui.

Aquí tan sons ens interessa el tema estricte de format de comunicacions I formats.

Per assegurar la qualitat de les dades I una mica la seguretat de les operacions disposarem de un string secret que serà al servidor I als clients.

El client afegirà automàticament 2 paràmetres,:

* t que te el temps en milisegons des de la epoch. Es per donar variabilitat al query
* hash que te el md5 de t+op+id ordenats alfabèticament primer id, desprès op I desprès t) + el secret. Si es volen mes paràmetres es poden afegir sense problemes.

El servidor rebrà el query, extreu les dades del $\_GET I calcula el hash. Compara I si no es igual que el rebut procedeix o be a no retornar res o a retornar “IR” únicament

Les respostes tenen següent format amb les línies separades per “\nl” :

* Línia 1 el Hash
* Línia 2 el temps en milisegons de l’epoch
* Línia 3 status de retorn (OK, ERROR)
* Línia 4 El codi de operació que hem tractat
* Línia 5 Lles respostes, sigui el missatge de error o noves dades etc. depenent de l’operació. Son en format CSV camps separats per “;”

El càlcul del Hash es semblant :

Es construeix el cos amb el temps + status `codi operació + resposta I es calcula el hash sobre això.

Els paràmetres habituals ha de ser:

* ip : Identifica la operació a executar
* id : Identifica l’objecte a operar. Normalment es la id de la base de dades
* terminal: Identifica elt erminal orígern de la operació a efectes de control en alguns casos com compres (saber quí ha fet la compra).

A la plantilla adjunta tan sols s’ha de subministrar una funció que donades la op, id i el terminal retorna un array [status, dades] a on les dades son la resposta.

Així doncs suposant que creem una funció que gestiona totes les. Operacions I reb els paràmetres op, id i terminal I retorna un array [status, dades] tenim un programa molt senzill que fa tot I manté la integritat de les comunicacions:

<?php

// cridat així: https://seguridadwebsimpo.pagaia.club/wp-content/QR\_Prova\_01.php?op=101&id=747&t=123456&hash=a0670ada2148a01e19506a80f1a27386d

// On op indica el qué s'està demanat. Per exemple 010 = Ha arrivat; 011 ha recollit obsequi; 020 Esmorzar 1er dia

// On id es l'identificador del participant. Un numero sequencial i ùnic que genera el gravity Forms

// VALIDACIO ENTRADA.

function valida($params, $secret, $alg){

$keys = array\_keys($params);

sort($keys);

$body = "";

foreach($keys as $key){

if($key != "hash"){

$body = $body . $params[$key];

}

}

$myhash = hash($alg, $body . $secret);

return ($myhash == $params["hash"]);

}

function buildAnswer($op, $data, $status, $alg, $secret){

$t = (string)microtime(); // Es podria fer servir time()

$answer = $t . "\n" . $status . "\n". $op . "\n" . $data;

$hash = hash($alg , $answer . $secret);

return $hash . "\n" . $answer;

}

// A aquesta funció gestionem les operacions. Canviar per el cas real.

function gestionaOp($op, $id, $terminal){

if ($op == "participants"){

$Resposta="000"; //

$PartNom= "Jordi Morell";

$status = "OK";

$data = $Resposta . ";". $PartNom . "\n";

}else {

$status = "ERROR";

$data = "La operació {$op} no està definida";

}

return[$status, $data];

}

// Main function. No s'ha de tocar

$secret = "asdjadskfjdaslkfj";

$alg = "md5";

$op=$\_GET['op'];

$id=$\_GET['id'];

$terminal = =$\_GET[‘terminal’];

if(valida($\_GET, $secret, $alg)){

list($status, $data) = gestionaOp($op, $id, $terminal);

echo buildAnswer($op, $data, $status, $alg, $secret);

}else{

echo "IR" ;

}

?>