Safestagram

**(스테가노그래피를 이용한 SNS 사진 보안 기술)**

**지도교수 서화정 교수님**

**팀장 1771164 유혜진**

**팀원 1771055 김정현**

**< 목차 >**

1. 표지 ---------------------------------------------- page 1
2. 목차 ---------------------------------------------- page 2
3. 프로젝트 개요 ----------------------------------- page 3
4. 주제 선정 배경 ---------------------------------- page 3
5. 주요 기능 및 시스템 구성 ---------------------- page 4
6. 개발시 요구된 기술 및 소프트웨어/하드웨어 소개 ---- page 5
7. 구현 결과 상세 소개 (사진 포함) ----- page6 – page19
8. 개발 일정 및 팀원별 담당 업무 --------------- page 20
9. 지원금 집행 내역 ------------------------------- page 21
10. 참고문헌 (도서, 논문, 웹사이트 등) ----------- page 21
11. 프로젝트 수행 후기---------------------------- page 22

**1. 프로젝트 개요**

- 개인 SNS 사진 무단 도용의 개인 정보 보안 문제 발생을 예방.

- 개인 SNS에 로그인 하면 사용자 정보가 담긴 secret.txt 파일이 관리자 컴퓨터에 생성되고, 사진을 업로드 할 시 이미지와 secret.txt가 함께 압축되어 관리자 컴퓨터에 저장, 관리자는 따로 관리자 페이지에서 압축 된 파일 압축 해제 가능.

- 타인이 사진을 무단으로 캡처해서 다른 곳에 업로드한 사실을 무단 도용 피해자가 인지

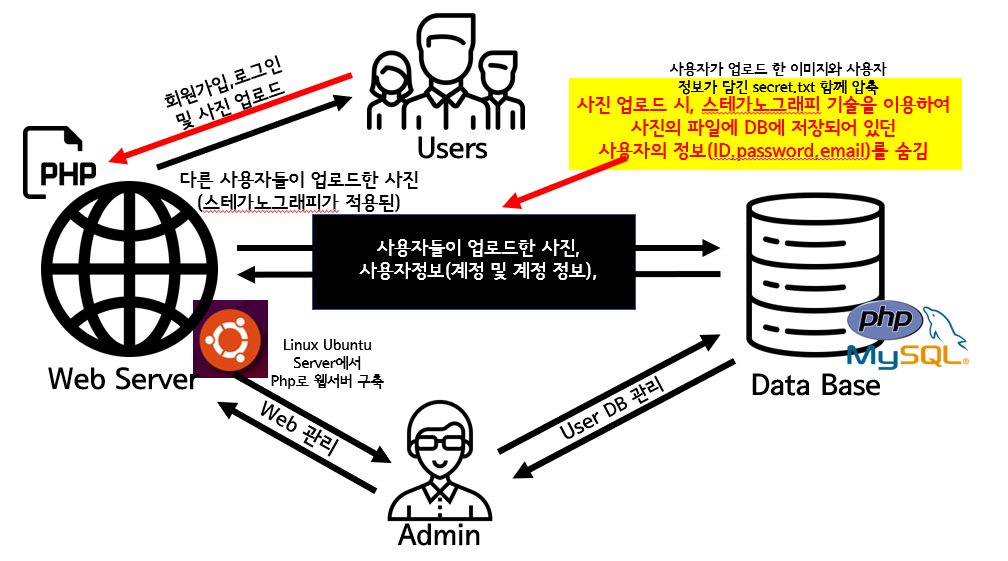
- 해당 사진을 SNS 관리자에게 신고하면, 관리자는 해당 이미지파일 압축 해제를 통해 사진 무단 유포자를 확인, 추후 처벌하는데 도움을 줄 수 있게 함.

**2. 주제 선정 배경**

사람들의 SNS 이용이 점점 더 증가하는 요즘, 사용이 증가함에 따라 많은 문제들도 함께 생겨나고 있음. 그 중 하나인 사진 무단 도용 피해를 예방하고 범인 추적에 간편하기 위해 이 프로젝트를 기획 함.

**3.  주요 기능 및 시스템 구성**

**3 – 1. 시스템 구성도**



**3 – 2. 주요 기능**

1. **Users**

- 가상의 웹 SNS페이지에 회원가입, 로그인이 가능

- 로그인 시 사용자가 알 수 없게 로그인 사용자의 정보를 받아와 데이터 파일이 작성됨.

- SNS board를 통해 다양한 사진을 업로드

- 사진 업로드 시 사용자가 알 수 없게 해당 사진을 업로드하는 사용자의 정보가 담긴 데이터 파일이 이미지에 숨겨진 채 업로드 됨.

- 다른 회원이 업로드한 사진도 함께 볼 수 있음

1. **Admin**

- 회원 관리 페이지를 통해 회원들의 정보와, 업로드 된 이미지들을 통합적으로 관리 가능

- 필요시에 따라 회원 또는 이미지를 관리자권한으로 삭제 가능(db)

- 이미지에 숨겨진 데이터 파일 복호화 가능 (숨겨진 데이터 확인)

**4. 개발시 요구된 기술 및 소프트웨어/하드웨어 소개**

1. SNS 형태를 가진 웹서버 구축

* 회원가입 시 MySQL DB에 회원가입 정보 저장
* 로그인 후 이미지 업로드시 게시판에 이미지와 업로드 한 사용자 아이디 게시
* 이미지 업로드 시 업로드 한 이미지 정보 MySQL DB에 저장
* 관리자 로그인 따로 가능하고 사용자 관리가 가능한 관리자 페이지 구축
* 관리자 페이지에서 사용자 삭제, 업로드 한 이미지 삭제, 암호화 된 이미지 복호화 가능

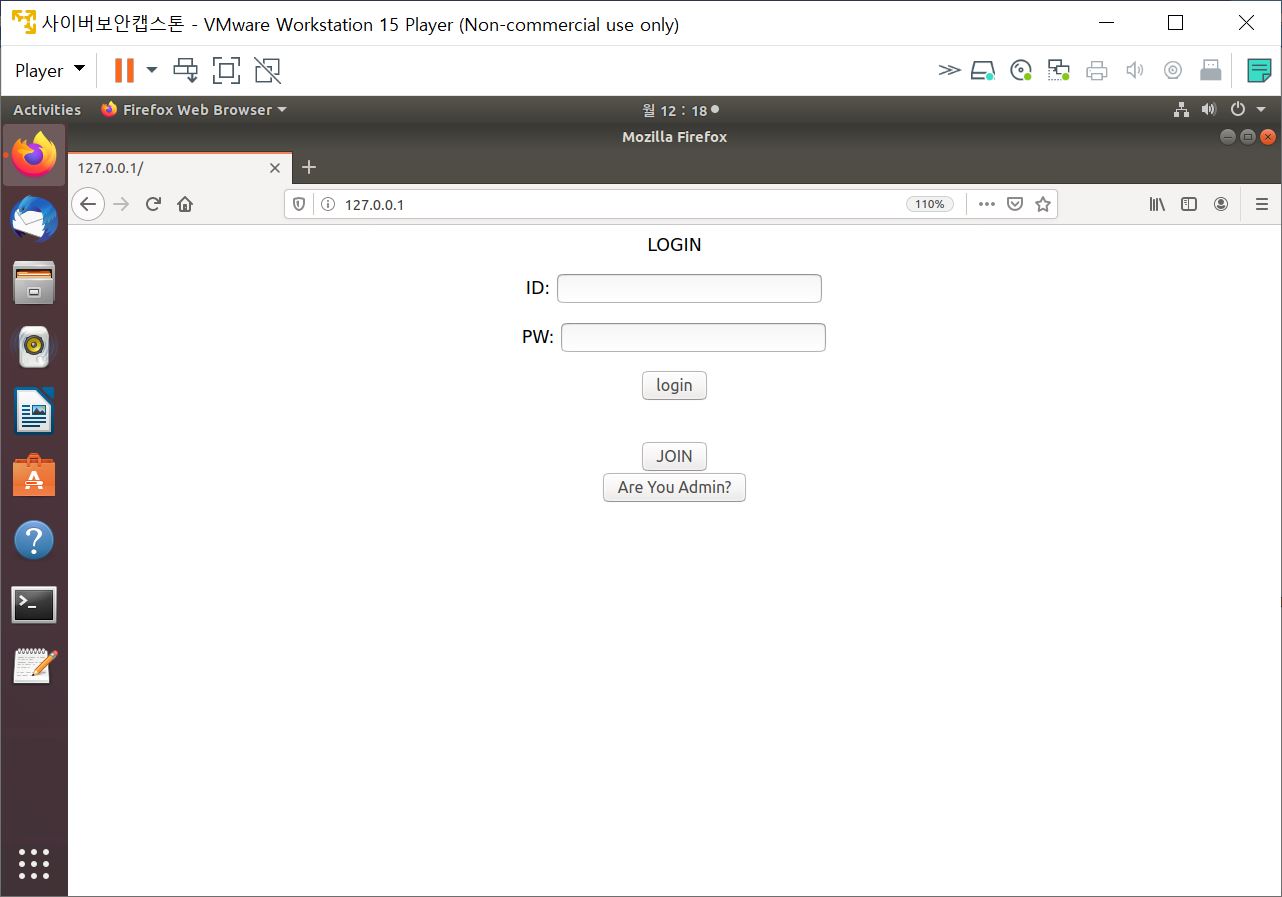
1. 사용자 및 관리자 계정, 업로드 된 이미지 저장하는 DB 구축 (MySQL)
2. PHP로 스테가노그래피 (이미지와 사용자 정보 txt 압축) 기술 삽입

- 스테가노그래피 : 사진, 음악, 동영상 등의 일반적인 파일 안에 데이터를 숨기는 기술

- php와 리눅스 함수 및 명령어를 이용하여 압축 및 압축 해제

1. 운영체제 : Linux Ubuntu
2. 언어 : PHP, HTML (웹)

1. **구현 결과 상세 소개**
2. 첫 메인 화면 (회원 로그인 페이지, index.html)



**<index.html 코드 부분 삽입>**

//get으로 id, pw입력 받아 login\_action.php로 넘김

<form method='get' action='login\_action.php'>

<p>ID: <input name="id" type="text"></p>

<p>PW: <input name="pw" type="password"></p>

<input type="submit" value="login">

</form>

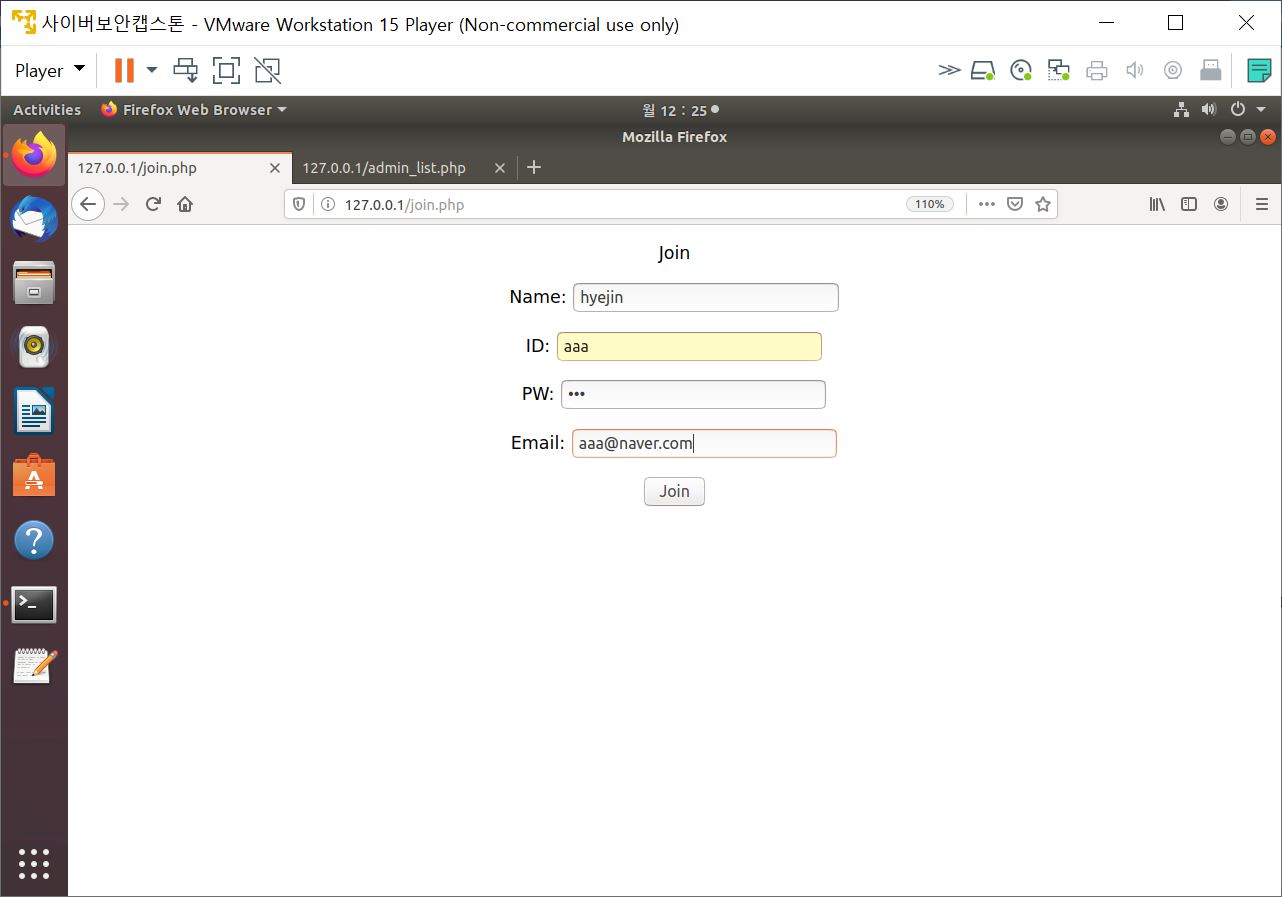
<br />

<button id="join" onclick="location.href='./join.php'">JOIN</button> //회원가입 페이지로 이동

<br />

<button id="admin" onclick="location.href='./admin\_login.php'">Are You Admin?</button> // 관리자 페이지로 이동

1. 메인 페이지의 JOIN 버튼을 누르면 회원가입페이지로 이동 (join.php)



**<join.php 코드 부분 삽입>**

//post형식으로 입력 후 값을 join\_action.php로 넘겨줌

<form method='post' action='join\_action.php'>

<p>Name: <input type="text" name="name"></p>

<p>ID: <input type="text" name="id"></p>

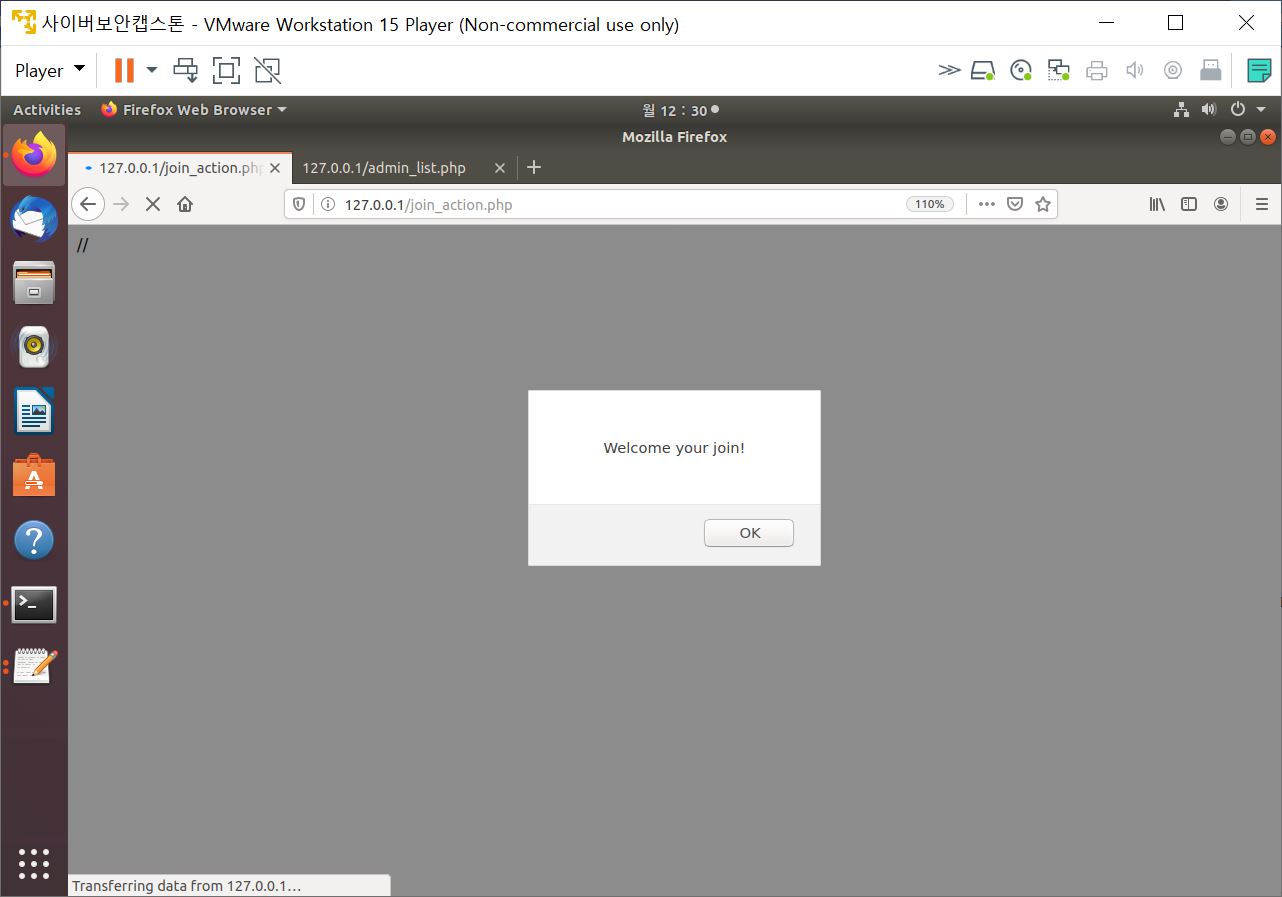
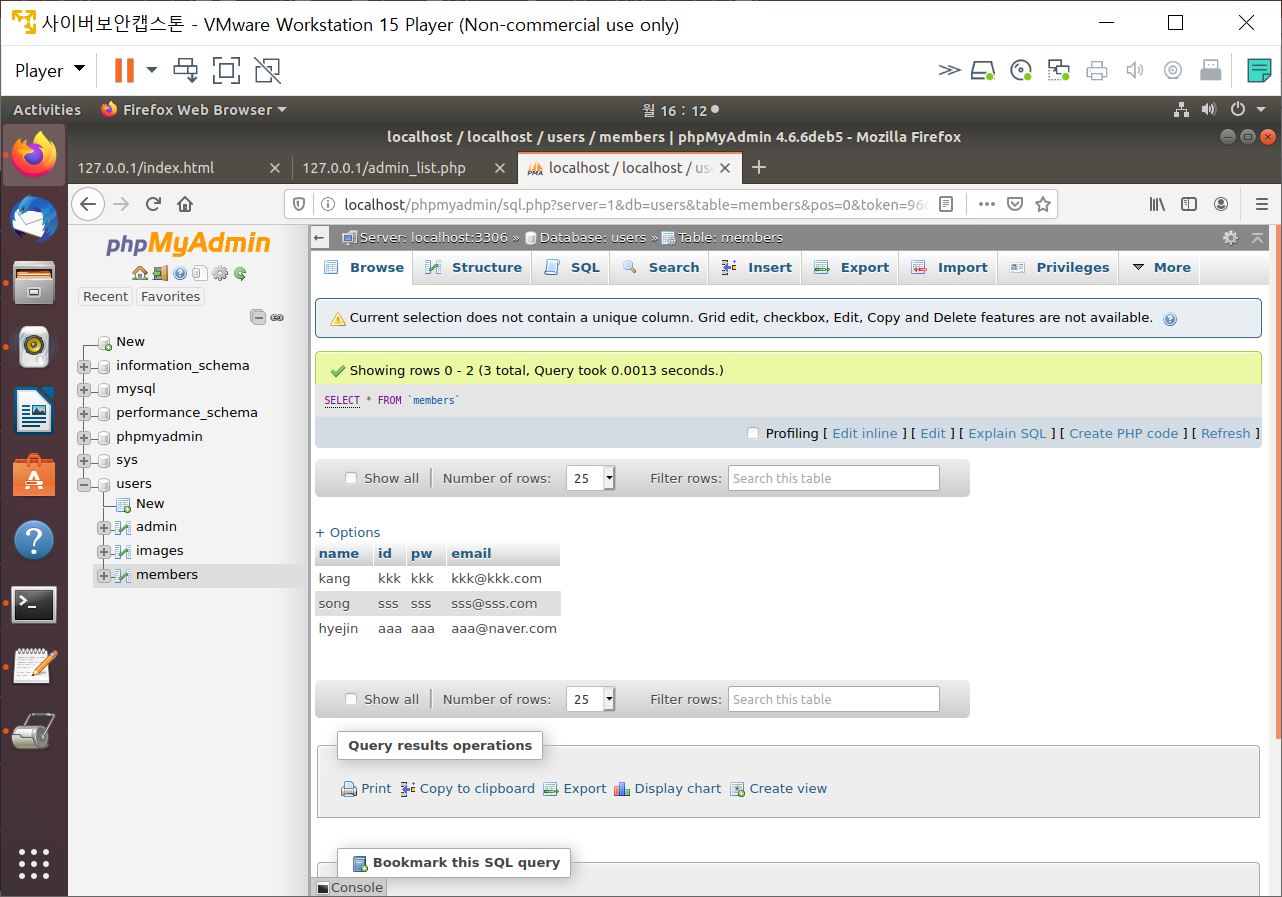
<p>PW: <input type="password" name="pw"></p>

<p>Email: <input type="email" name="email"></p>

<input type="submit" value="Join">

</form>

1. 알맞게 회원가입정보를 입력하면 회원가입완료 알림창이 뜸 (join\_action.php)



**<join\_action.php 코드 부분 삽입> < 위) 회원가입성공시 mysql db >**

$connect = mysqli\_connect('localhost','superuser','0000','users') or die("fail");

$name=$\_POST[name];

$id=$\_POST[id];

$pw=$\_POST[pw];

$email=$\_POST[email];

$query = "insert into members (name, id, pw, email) values ('$name', '$id', '$pw', '$email')"; //post형식으로 입력받은 값 insert

$result = $connect->query($query);?>

<?php

if($result) { // db에 회원정보 입력 성공 시

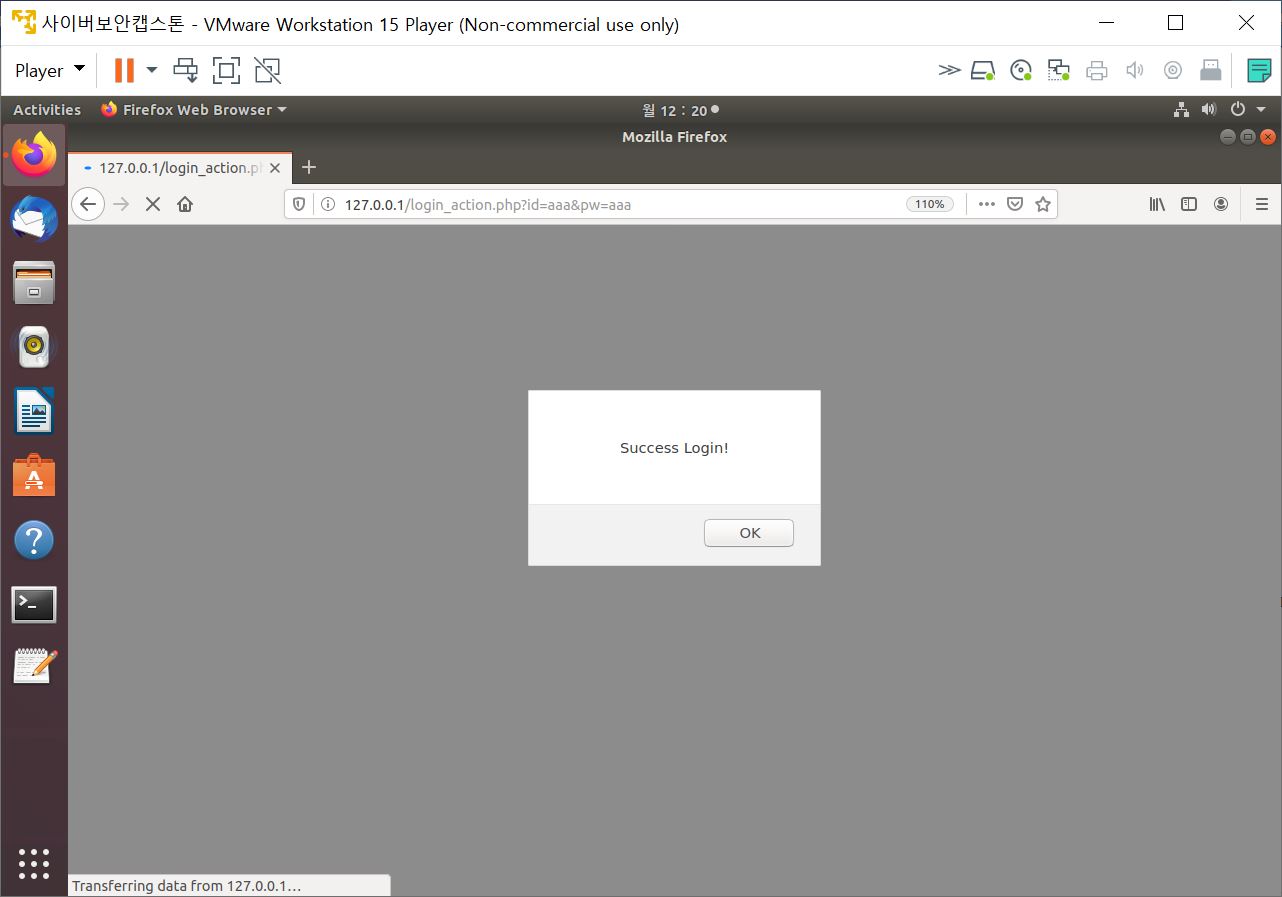
?> <script>

alert('Welcome your join!');

location.replace("./index.html");

</script>

1. 다시 메인 페이지로 이동해서, 가입한 id, password 이용 로그인 진행(login\_action.php)



**<login\_action.php 코드 부분 삽입>**

$connect = mysqli\_connect("localhost","superuser","0000","users") or die("fail");

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

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

$query = "select \* from members where id='$id'"; //입력받은 id의 회원 쿼리에서 검색

$result = $connect->query($query);

if(mysqli\_num\_rows($result)==1){

$row=mysqli\_fetch\_assoc($result);

// 해당 id와 password가 일치하는지 확인

if($row['pw']==$pw){

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

if(isset($\_SESSION['id'])){

?>

<?php

// 이미지에 숨길 회원 정보파일 받아오기

$DOCUMENT\_ROOT = $\_SERVER['DOCUMENT\_ROOT']; //현재 경로(/var/www/html) 변수로 저장

$upload\_dir = "$DOCUMENT\_ROOT/uploads/sk/"; // 업로드할 디렉토리 경로 변수로 저장

if(!is\_dir($upload\_dir)){ // 디렉토리 유무 확인 후

mkdir($upload\_dir); // 변수의 디렉토리 생성

}

$myfile = fopen("$DOCUMENT\_ROOT/uploads/sk/secret.txt","w") or die("Unable to open file!"); //해당 디렉토리에 비밀txt파일 쓰기형식으로 create,open

$txt = $row['name']."\t".$id."\t".$pw."\t".$row['email']."\n"; // 로그인한 사용자의 정보

fwrite($myfile, $txt); //쓰기

fclose($myfile);

exec("cd /var/www/html/uploads && zip -r sk.zip sk"); // 해당 경로로 이동 후, secret.txt가 들어있던 sk디렉토리 압축

?>

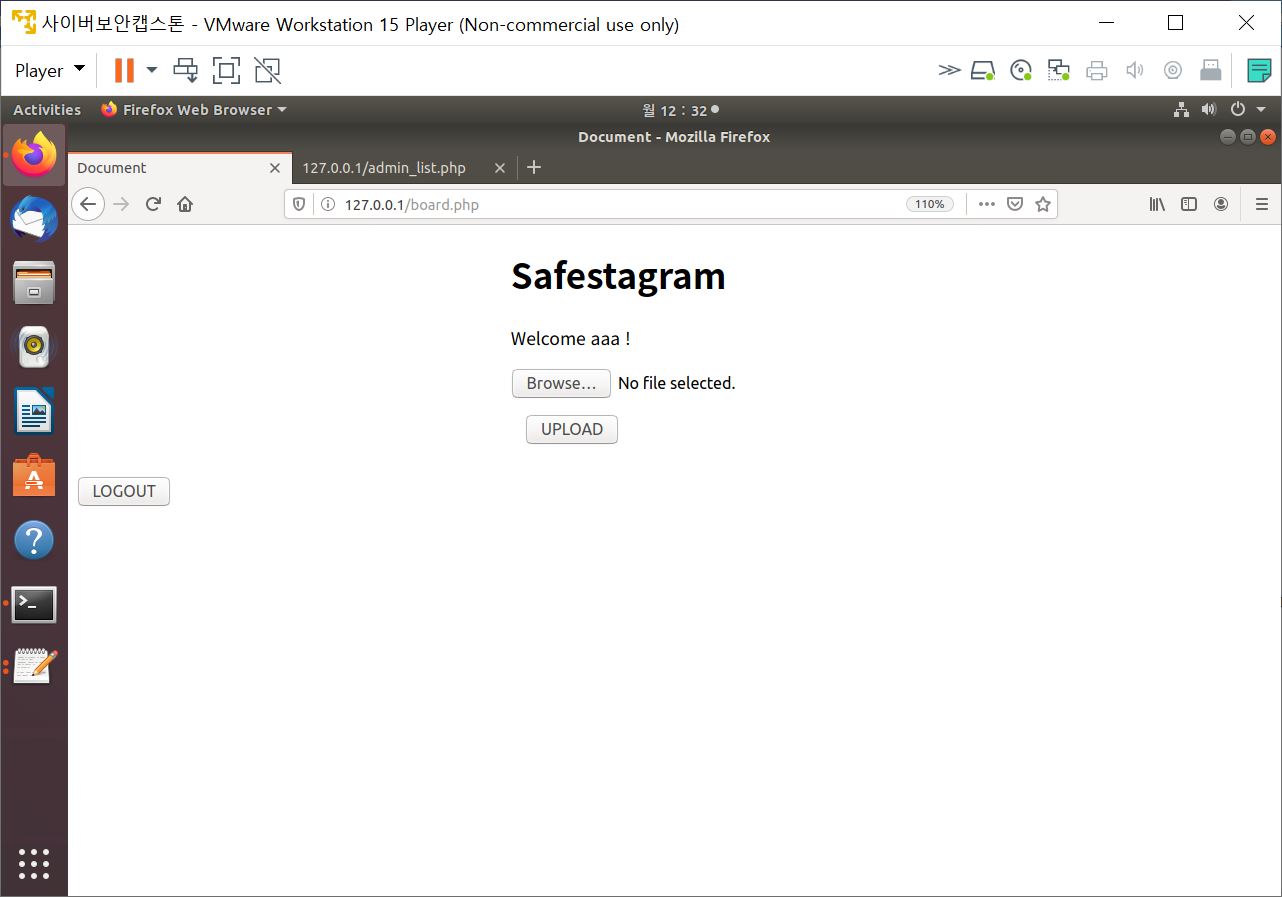
<script>

alert("Success Login!");

location.replace("./board.php"); //login성공시 게시판 페이지로 이동

</script>

1. 로그인 후 게시판 화면 (board.php)



**<board.php 코드 부분 삽입>**

<div style="width: 300px; margin:0 auto;">

<h1>Safestagram</h1>

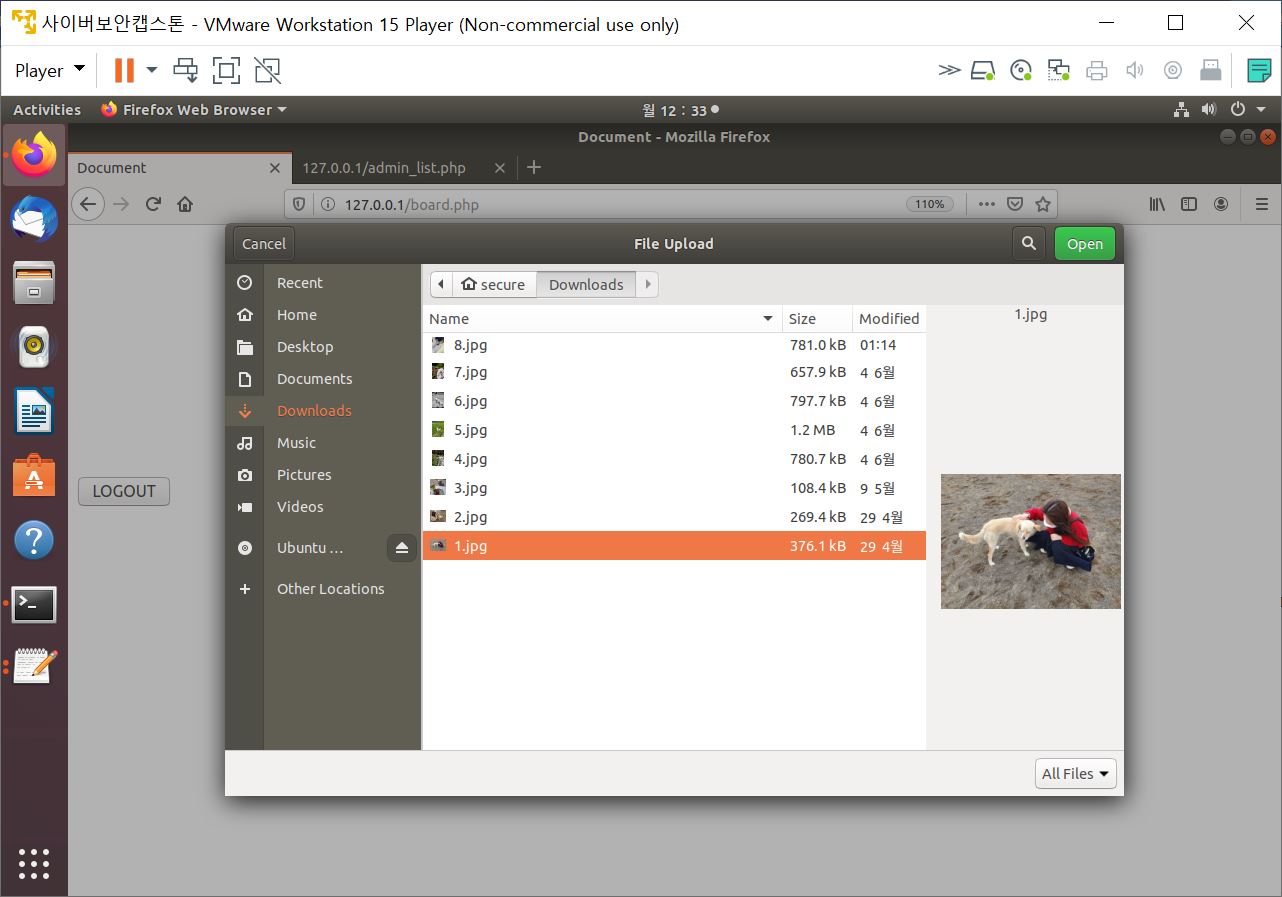
<?php

session\_start();

echo "Welcome ". $\_SESSION['id']." !";

?><p>

1. Browse… 버튼을 눌러 업로드 할 이미지 선택



**<board.php 코드 부분 삽입>**

//이미지 업로드를 위한 툴

<form action="new\_upload.php" method="post" enctype="multipart/form-data">

<div>

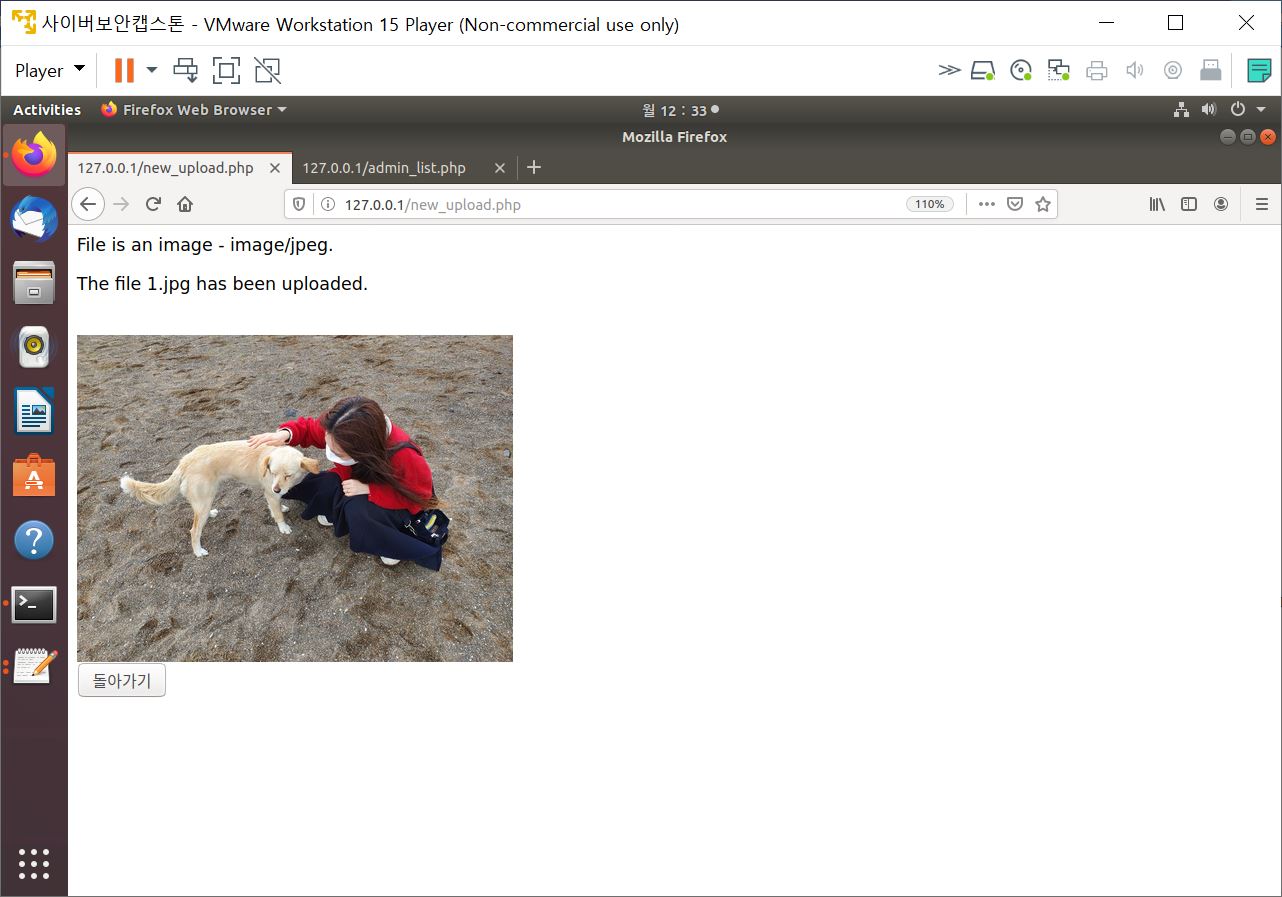
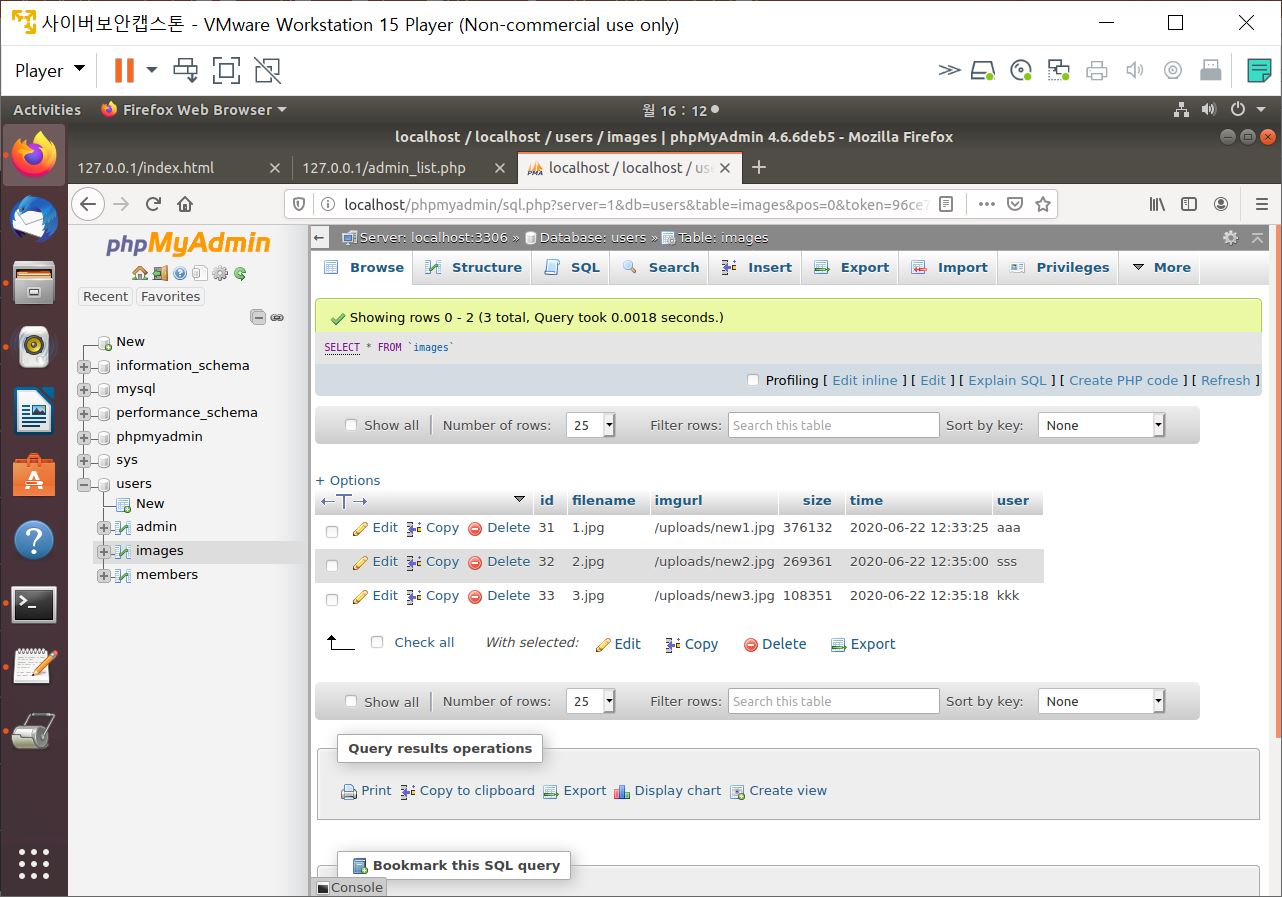
<input type="file" name="fileToUpload" id="fileToUpload"> </div>

<input type="submit" value="UPLOAD" name="submit" style="margin: .9em">

</form>

</div>

1. 이미지 업로드 시 업로드 성공 화면(new\_upload.php)



**<new\_upload.php 코드 부분 삽입> < 위) 이미지 업로드 성공시 mysql db >**

if (move\_uploaded\_file($\_FILES["fileToUpload"]["tmp\_name"], $target\_file)) {

//사용자로부터 업로드 된 이미지에 정보 숨기기

exec("cd /var/www/html/uploads && cat $img\_file sk.zip > $new\_img "); //사용자 로그인시 만들어졌던 사용자정보 비밀 파일과 이미지 저장

//비밀 텍스트 파일, 디렉토리, zip파일 삭제

exec("cd /var/www/html/uploads/sk && rm secret.txt");

exec("cd /var/www/html/uploads/sk && rm sk.zip");

exec("cd /var/www/html/uploads && rmdir sk");

session\_start();

$filename = $\_FILES["fileToUpload"]["name"];

$imgurl ="/uploads/". $new\_img; // 암호화된 이미지 파일 경로로 저장

$size = $\_FILES["fileToUpload"]["size"];

$user = $\_SESSION['id'];

//mysql db 연결

$conn = mysqli\_connect('localhost', 'superuser', '0000', 'users');

$sql = "insert into images(filename, imgurl, size, user) values('$filename','$imgurl','$size','$user')";

mysqli\_query($conn,$sql);

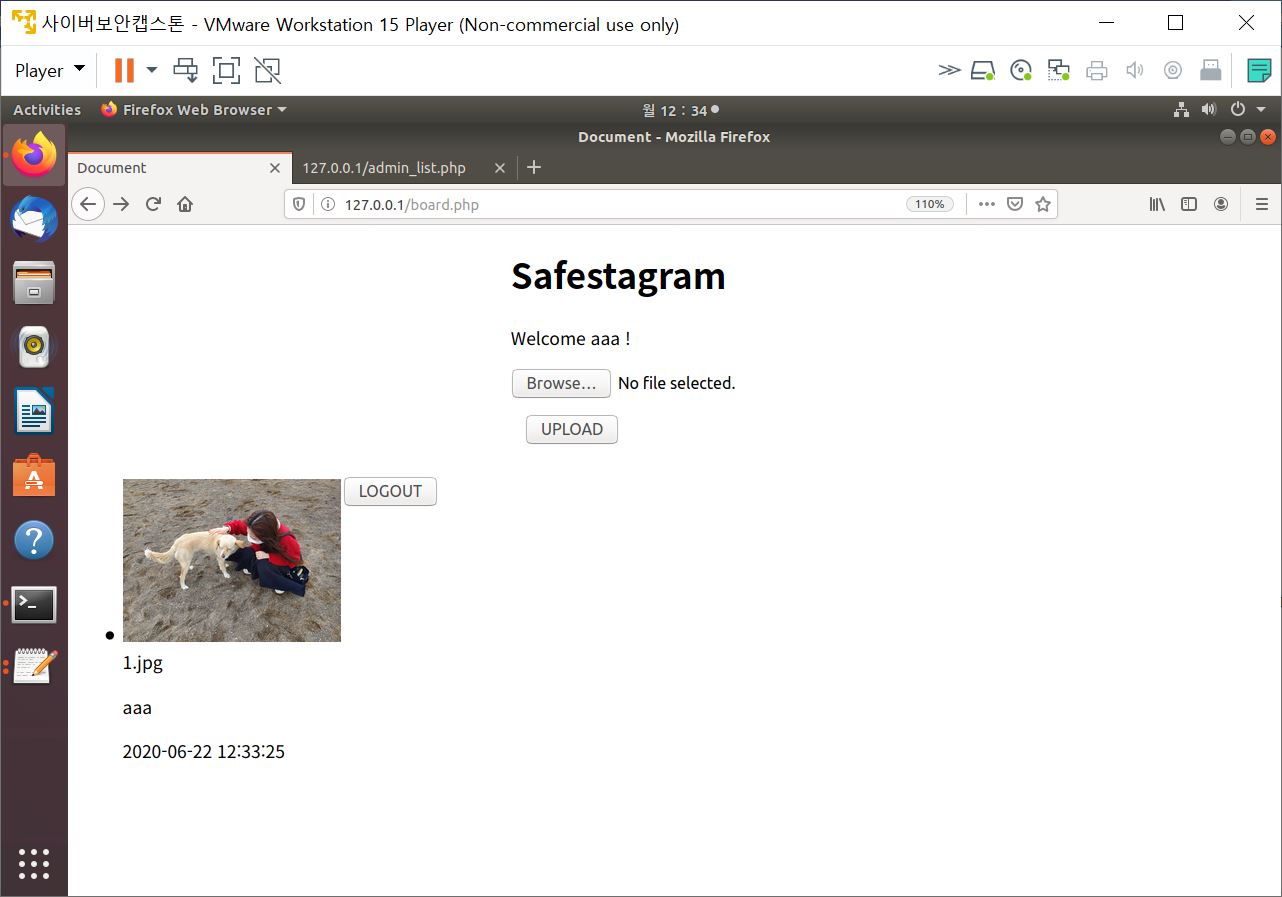
mysqli\_close($conn);

echo "<p>The file ". basename( $\_FILES["fileToUpload"]["name"]). " has been uploaded.</p>";

echo "<br><img src=/uploads/". basename( $\_FILES["fileToUpload"]["name"]). " width=400>"; //업로드된 이미지 보여줌

echo "<br><button type='button' onclick='history.back()'>돌아가기</button>";

1. 돌아가기 버튼을 누르면 이미지가 올바르게 업로드된 게시판 화면이 보여짐(board.php)



**<board.php 코드 부분 삽입>**

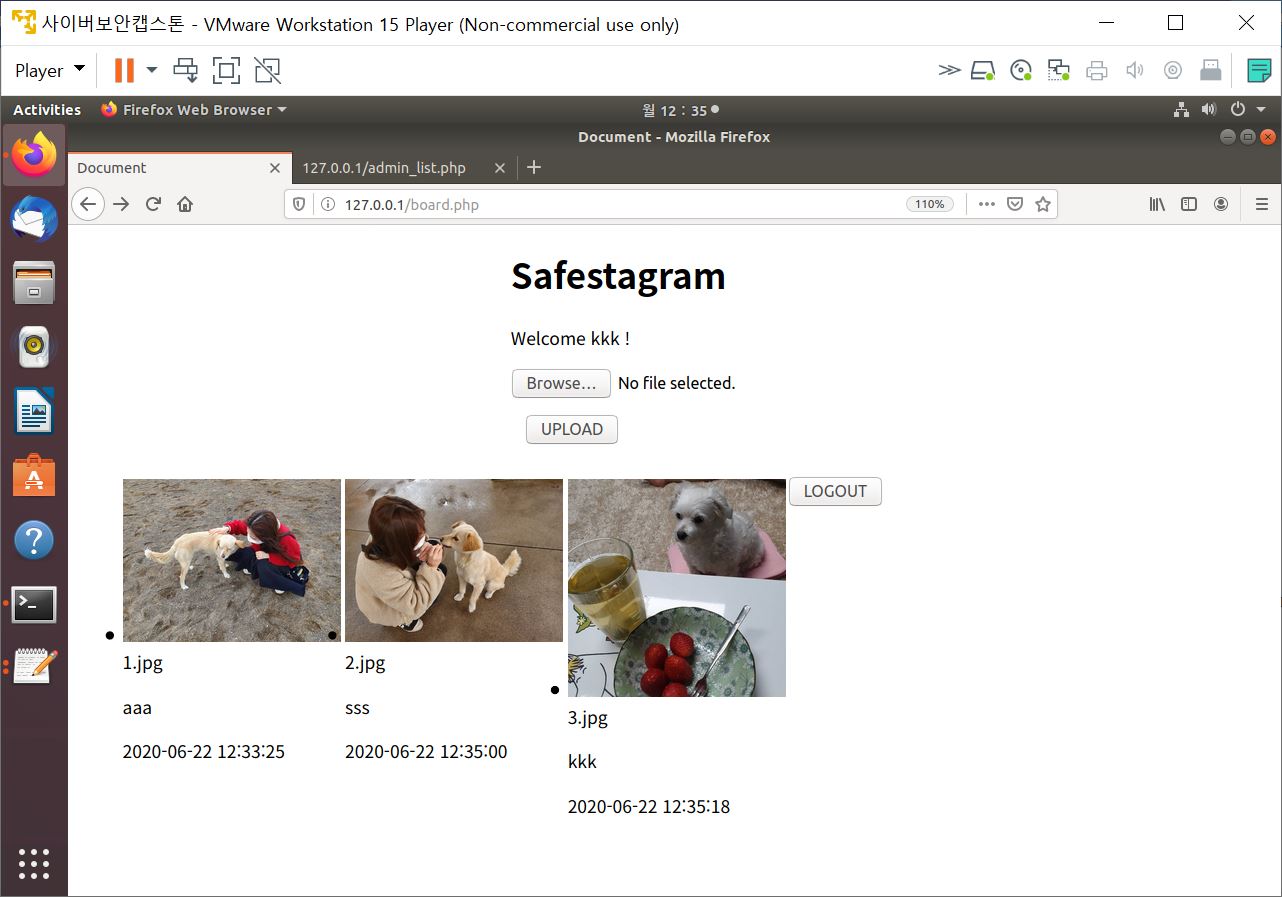
// 해당 이미지, 업로드한id, 올린 시간

echo '<li style=\'float:left; margin: 2px;\'>';

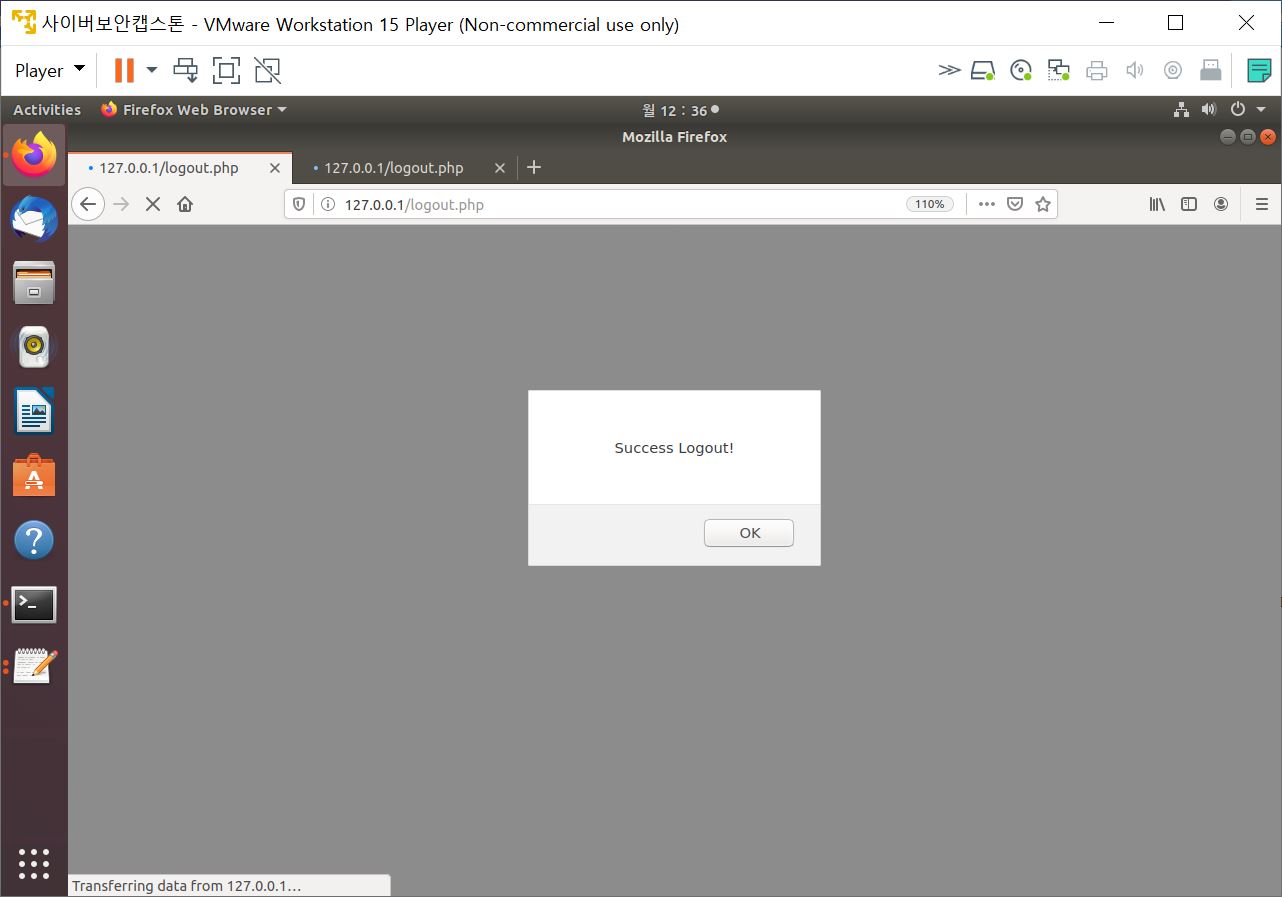
echo '<img src='.$data['imgurl'].' width=200><br>';

echo ($data['filename']);

1. 다른 계정들로도 로그인 후 이미지들을 업로드 한 화면(board.php)



1. 로그아웃(logout.php)



**<board.php 코드 부분 삽입>**

$result = session\_destroy(); // 현재 세션 취소(로그아웃)

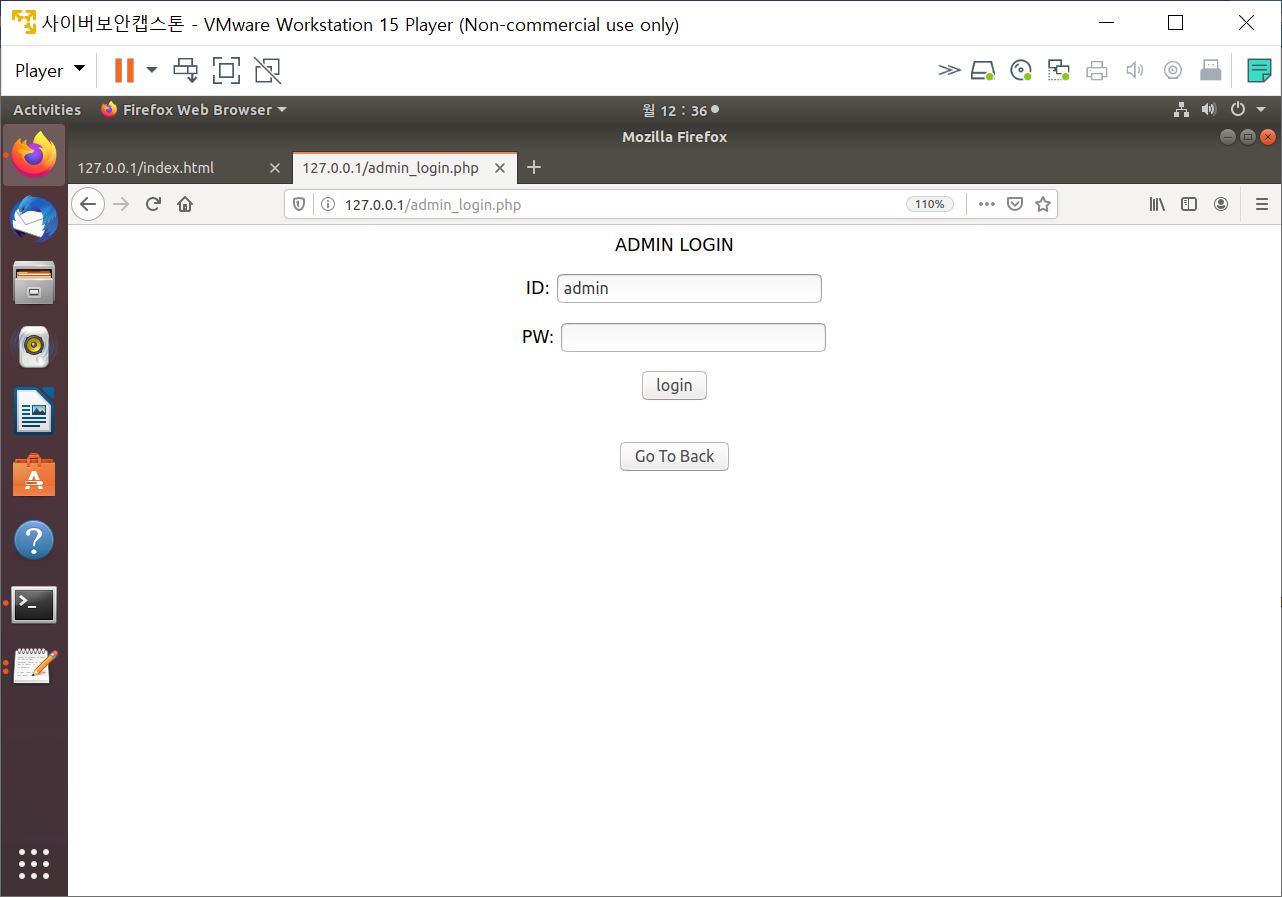
if($result) {

?> <script>

alert("Success Logout!");

1. 첫 메인 화면에서 Are You Admin? 버튼을 눌러 관리자 로그인 페이지로 이동

(admin\_login.php)



**<admin\_login.php 코드 부분 삽입>**

//get으로 id, pw입력 받아 admin.php로 넘김

<form method='get' action='admin.php'>

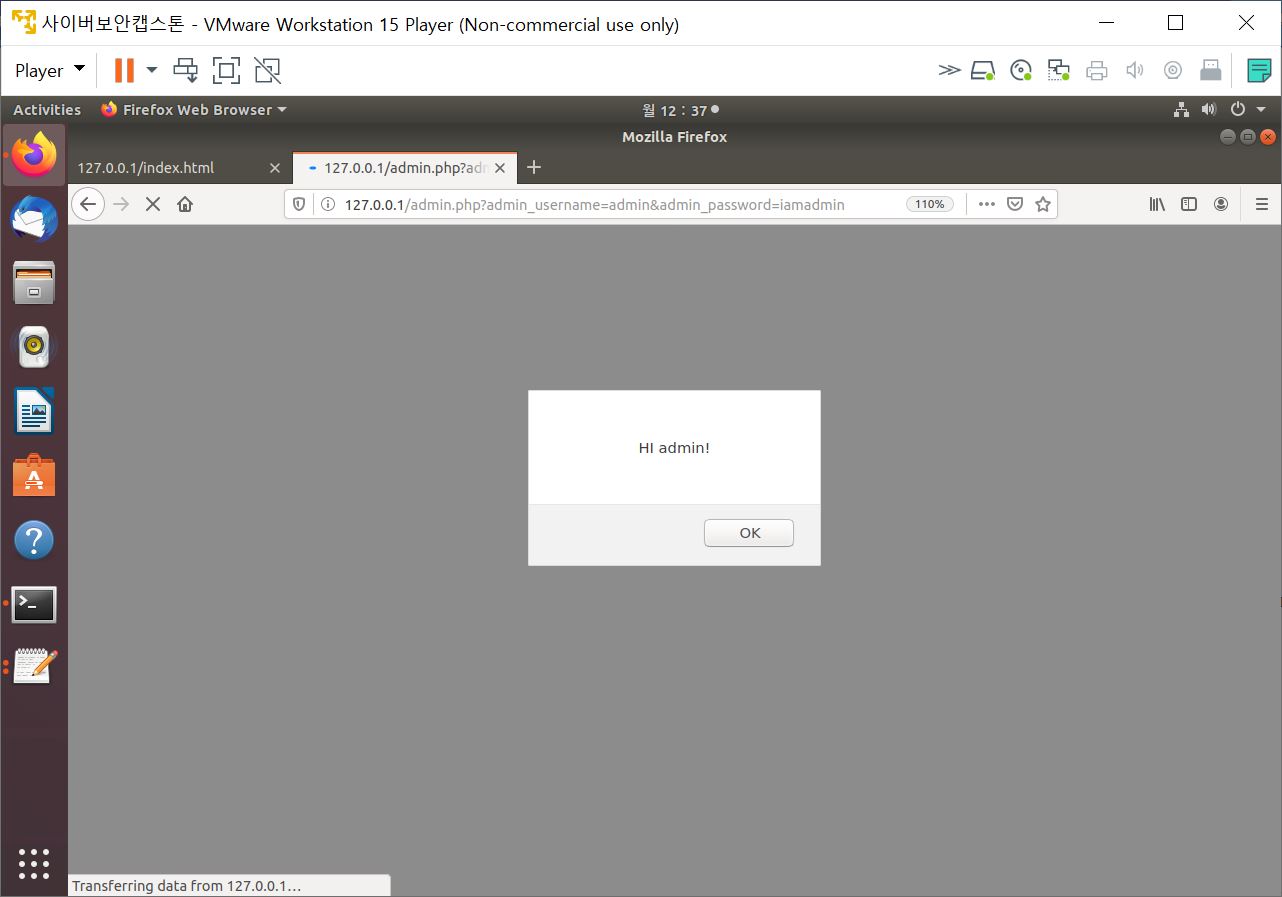
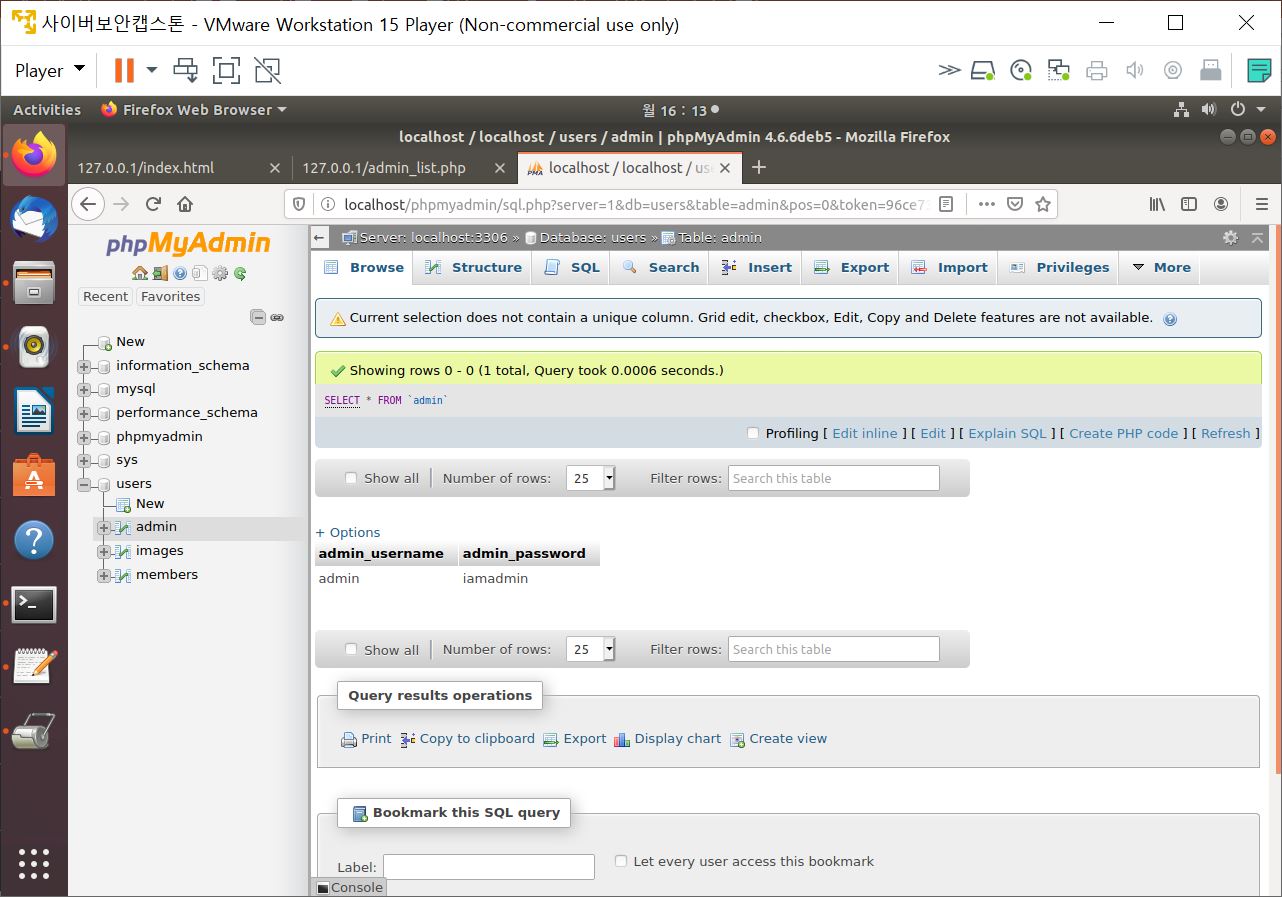
<p>ID: <input name="admin\_username" type="text"></p>

<p>PW: <input name="admin\_password" type="password"></p>

<input type="submit" value="login">

</form>

1. 저장되어있던 특정 관리자 계정으로 로그인 성공 시 (admin.php)



**<admin.php 코드 부분 삽입> < 위) db에 미리 설정해놓은 관리자 계정>**

$connect = mysqli\_connect('localhost', 'superuser', '0000', 'users') or die ("fail");

$admin\_username = $\_GET['admin\_username'];

$admin\_password = $\_GET['admin\_password'];

$query = "select \* from admin where admin\_username='$admin\_username'"; // 해당 username과 password 일치하는 쿼리 저장

$result = $connect->query($query);

if(mysqli\_num\_rows($result)==1){

$row=mysqli\_fetch\_assoc($result);

// 일치시 관리자 페이지로 이동

if($row['admin\_password']==$admin\_password){

$\_SESSION['admin\_username']=$admin\_username;

if(isset($\_SESSION['admin\_username'])){

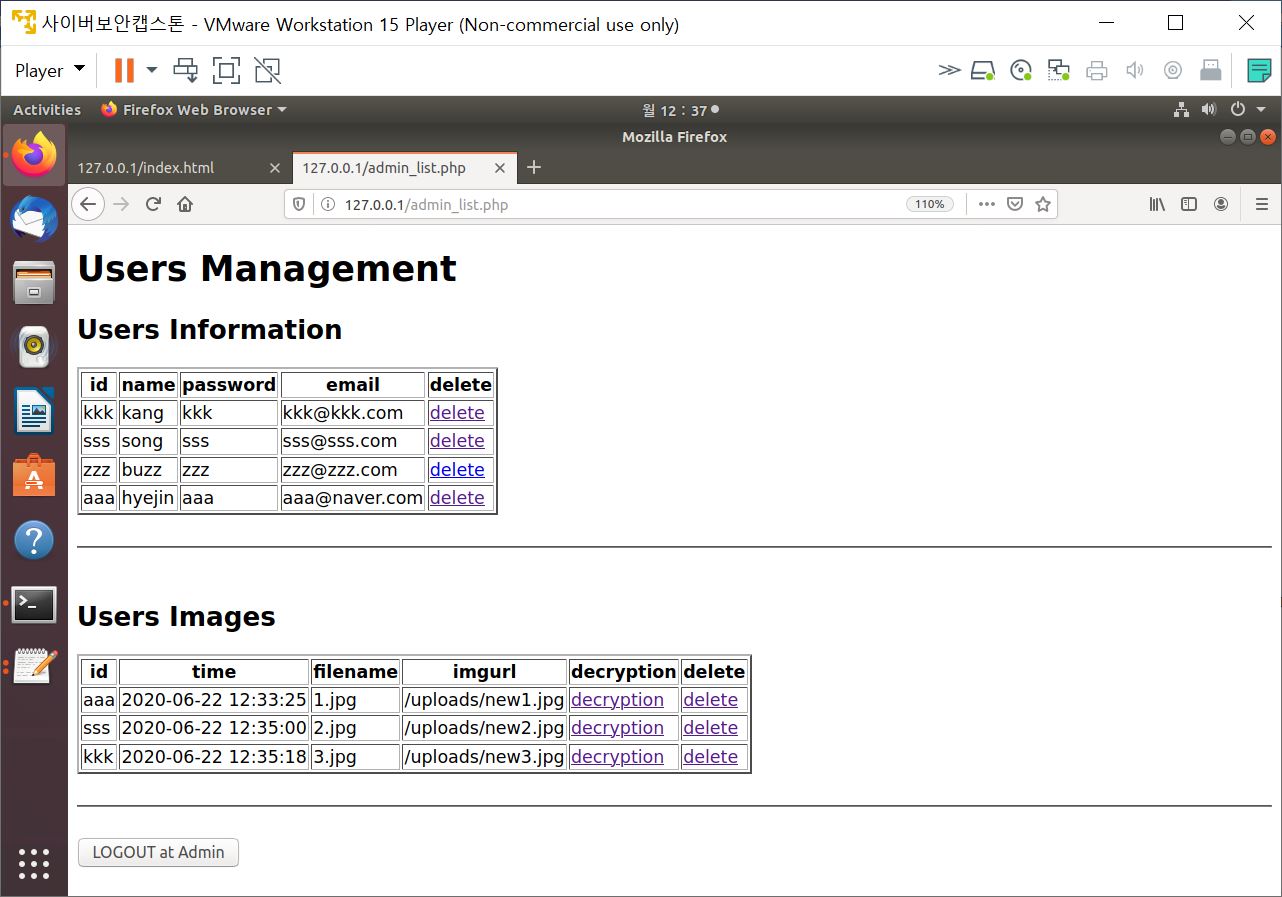
?> <script>

alert("HI admin!");

location.replace("./admin\_list.php");

</script>

1. 관리자의 회원관리 페이지(admin\_list.php)



**<admin\_list.php 코드 부분 삽입>**

//db에 연결 (회원 정보)

<?php

$conn = mysqli\_connect('localhost', 'superuser', '0000', 'users');

$query = "SELECT \* FROM members";

$result = mysqli\_query($conn, $query);

while($row = mysqli\_fetch\_array($result)){

$id = $row[1];

$name = $row[0];

$pw = $row[2];

$email = $row[3];

?>

//회원 정보 표에 삽입

<tr>

<td> <?php echo $id; ?> </td>

<td> <?=$name ?></td>

<td> <?=$pw ?> </td>

<td> <?=$email ?> </td>

<td> <a href="delete.php?del=<?=$id?>">delete</a></td> //회원정보 삭제php로 이동 버튼(해당id도넘겨줌)

</tr>

</table><br><hr><br>

<h2> Users Images</h2> // 업로드된 이미지

db에 연결 (이미지 정보)

<?php

$conn = mysqli\_connect('localhost', 'superuser', '0000', 'users');

$query = "SELECT \* FROM images";

$result = mysqli\_query($conn, $query);

while($row = mysqli\_fetch\_array($result)){

$user = $row[5];

$time = $row[4];

$filename = $row[1];

$imgurl = $row[2];

?>

//이미지 정보 표에 삽입

<tr>

<td> <?php echo $user; ?> </td>

<td> <?=$time ?> </td>

<td> <?=$filename ?> </td>

<td> <?=$imgurl ?> </td>

<td> <a href="decryption.php?dec=<?=$imgurl?>">decryption</a></td> //복호화php로 이동(해당이미지url넘겨줌)

<td> <a href="delete\_img.php?del=<?=$imgurl?>">delete</a></td> //이미지 삭제 php로 이동 (해당이미지url넘겨줌)

1. Delete 버튼을 누를 시 해당 회원 정보 또는 이미지 정보 db가 삭제됨

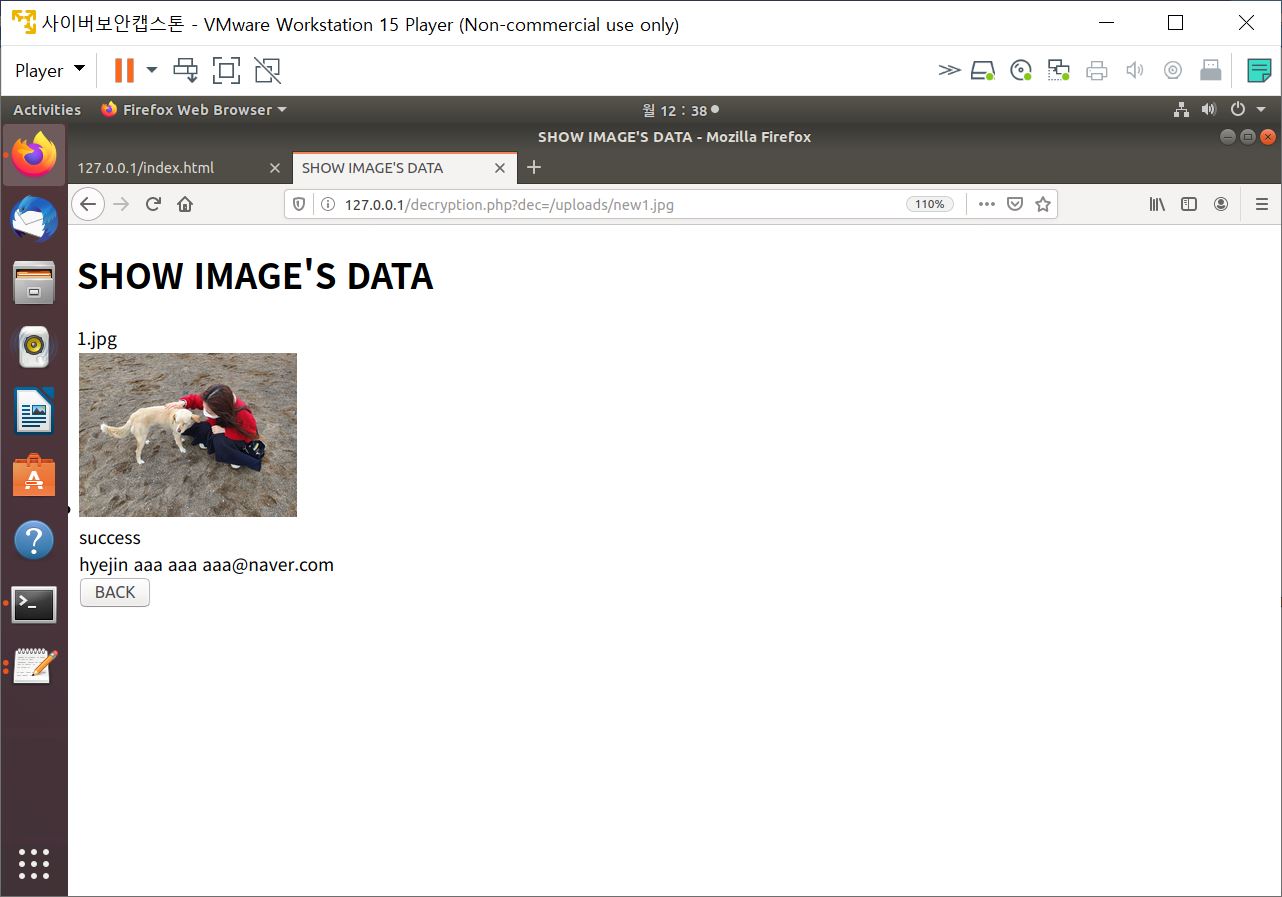
**<delete.php 코드 부분 삽입>**

$conn = mysqli\_connect('localhost', 'superuser', '0000', 'users');

$delete\_id = $\_GET['del'];

$query = "DELETE FROM members WHERE id='$delete\_id'"; //admin\_list.php에서 넘겨준 해당 id 값과 일치하는 회원 쿼리 삭제

1. Decryption 버튼을 누를 시 해당 이미지에 숨겨져 있던 업로드한 회원의 정보가 복호화되며 출력



**<decryption.php 코드 부분 삽입>**

//db에 연결 (회원 정보)

<?php

$conn = mysqli\_connect('localhost', 'superuser', '0000', 'users');

if(mysqli\_connect\_error()){

echo "Connect Error! ".mysqli\_connect\_error();

}

//이전 admin\_list.php에서 넘겨준 해당 이미지 url값 변수로 저장

$dec\_img= $\_GET['dec'];

$query = "SELECT \* FROM images WHERE imgurl='$dec\_img'"; //해당 url과 일치하는 이미지 쿼리 검색

$result = mysqli\_query($conn, $query);

while($data = mysqli\_fetch\_array($result)){

$new = "new" . $data['filename']; //암호화된 이미지파일이름으로

echo $data['filename'].'<br>' ;

echo '<li style=\'float:left; margin: 2px;\'>';

echo '<img src='.$dec\_img.' width=200><br>';

if (exec("cd /var/www/html/uploads && unzip $new")){ // 이미지가 있는 디렉토리로 이동후 압축품(복호화)

echo "success".'<br>';

readfile("uploads/sk/secret.txt"); } //이미지에 넣어둔 비밀 메시지가 들어있는 텍스트값 읽어옴

1. **개발 일정 및 팀원 별 담당 업무**

**6-1 개발 일정**

|  |  |
| --- | --- |
| **주차** | **목표** |
| **1~3주차** | 주제 선정 및 초기 제안서 작성 |
| **4주차** | Linux 개발 환경 셋팅 (VMware+ubuntu 설치) |
| **5주차** | 웹서버 환경 설정 |
| **6주차** | 웹서버 형태 구축( 로그인, 로그아웃, 회원가입 기능 구현) |
| **7주차** | MySQL 과 php 연동 |
| **8주차** | Php 이미지 업로드, DB에 이미지 저장 |
| **9주차** | 중간발표 |
| **10주차** | 이미지 업로드 오류 해결 |
| **11주차** | 관리자 페이지 생성 및 DB 구축 |
| **12주차** | 관리자 페이지 보완(사용자 리스트 관리 및 이미지 관리), 스테가노그래피 구현 |
| **13주차** | 스테가노그래피 기술 구현 |
| **14주차** | 서비스 동작 미비점 및 오류 보완, 최종 테스트 |
| **15주차** | 최종발표 |

**6-1 팀원 별 담당 업무**

**- 유혜진**

1. 웹서버 제작(회원가입, 로그인, 로그아웃, 이미지 업로드, 게시판, 관리자 로그인, 관리자 페이지)

2. Php와 MySQL 연동 및 MySQL DB 구축

3. 이미지 업로드 구현

4. 스테가노그래피 구현

**- 김정현**

1. 이미지 업로드 구현

2. 스테가노그래피 구현

1. **지원금 집행 내역**

|  |  |  |  |
| --- | --- | --- | --- |
| **날짜** | **사용목적** | **사용처** | **금액(내역)** |
| 20.05.12 | 회의비 | 한성대 연구관 그라지에 | 6,600  (아이스아메리카노L+샷추가 X 2잔) |
| 20.05.26 | 스타벅스 삼선교점 | 11,200  (아이스아메리카노V+콜드브루V) |
| 20.06.03 | 카페 코피티암 연신내점 | 12,600  (아이스아메리카노L+샷추가 X 2잔) |

**8. 참고문헌(도서, 논문, 웹사이트 등)**

- ubuntu 개발 환경 설치

<https://ubuntu.com/tutorials/install-and-configure-apache#3-creating-your-own-website>

- ubuntu 웹서버 개발을 위한 환경 구축

<https://nan491.tistory.com/entry/Ubuntu-%EC%9B%B9-%EC%84%9C%EB%B2%84Web-Server-%EC%84%A4%EC%B9%98%ED%95%98%EA%B8%B0>

- php 각 기능 구현

<https://www.w3schools.com/php>

- 리눅스 내의 스테가노그래피 구현

<https://www.ostechnix.com/hide-files-inside-images-linux/>

1. **프로젝트 수행 후기**

|  |
| --- |
| IT융합공학부 1771055 김정현 |
| 힘든 부분이 많았지만 그래도 정말 보람찬 한 학기였습니다. 캡스톤을 진행하면서 여러가지 자료를 찾아보고 적용해 보면서 많은 것을 배우게 됐고 리눅스, php, MySQL에 대하여 더 자세히 알 수 있었습니다. 어려움도 많았지만 오류가 생기는 부분을 해결해 나가고 프로그램이 진행되는 걸 보면서 보람을 많이 느꼈습니다. 다만 진행을 할 수록 내가 이 부분에 대해서 지난 3년동안 더 열심히 공부했더라면 더 쉽게 해내지 않았을까 지금보다 더 높은 퀄리티의 프로젝트를 제작할 수 있지 않았을까 하는 아쉬움이 많이 남습니다. 웹서버 제작과 DB연결을 포함한 많은 부분에서 유혜진 팀원이 정말 고생을 많이 했는데 그러한 부분에서 많은 도움이 되지 못한 것 같아 아쉬움이 많이 남습니다. 아직 미숙한 점이 많은 프로젝트인데 더 공부하여 부족한 부분을 보완해 더 나은 프로젝트로 발전시키고 싶습니다. |
| IT융합공학부 1771164 유혜진 |
| 얻는게 많았던 만큼 아쉬움도 많이 남는 캡스톤디자인 수업이었습니다. 처음에 아이디어 선정은 생각보다는 어렵지 않게 진행하였으나, 제 욕심과 달리 구현하기위해서는 한번도 다루지 않았던 부분들도 다뤄야했고, 기능 구현보다는 구현하는 과정에서의 오류해결과정에 시간을 너무 많이 소요한 것 같아 아쉬운 마음이 큽니다.  교수님께서 중간발표에 조언해주신 회원가입시 비밀번호의 보안성 강화, 또 관리자 로그인에서의 보안성 강화, 또 스테가노그래피로 사용자의 기본 정보 이외에 업로드한 IP주소 입력 등, 더욱 욕심났던 부분들은 많았으나, 오류들에 부딪치고 해결을 하다보니 정말 기본적인 기능 구현을 한 작품을 완성한 것 같아 아쉬움이 남습니다.  그러나 이를 계기로 리눅스 시스템과 웹서버에 대해서 더욱 공부하게 되는 계기가 되었고, 프로그래밍에 대해 자신이 없었는데 저의 부분적인 강점이 생긴 것 같아 이에 대해 더 공부해보고 싶습니다. 또 기회가 된다면 현재 프로젝트에 아쉬움이 남았던 부분들을 보태어 더욱 좋은 시스템을 구현 해보고 싶습니다.  마지막으로 문제가 생겼을 때 연락을 드리면 언제든 조언을 해주셨던 서화정 지도교수님과, 각 오류와 난제 해결을 위해 지난 4개월간 함께 해준 팀원 정현이에게 큰 감사드립니다. |