2015년 정보보호학과 졸업작품

SEED암호를 이용한 보안 웹 서버

팀명 : Ghost Service 24

팀장 : 박철오(10)
팀원 : 전인혁(10)
양유빈(10)
김지현(12)
김은지(11)
담당교수 : 양환석 교수님

2015.06
중부대학교 정보보호학과

<목차>

요약

2014년 5월 개인정보 유출로 곤혹을 치른 토니모리의 경우 홈페이지 해킹으로 50만 건의 개인정보가 유출됐고, 이보다 앞서 지난 4월 스킨푸드 는 홈페이지 해킹으로 개인정보가 유출됐다고 밝힌 바 있다. 이와 관련 기업들의 웹사이트 관리가 허술하다는 지적이 일고 있다. 인터넷서점 예스24의 경우 지난 18일 악성코드 유포지로 활용된 바 있고, 같은 날 조선일보사 계열인 디지털조선일보 보도부 브랜드대상 사무국 홈페이지의 일부 페이지가 악성코드 유포지로 악용된 바 있다. 특히 공격자들은 주말을 이용해 악성코드를 삽입한 후, 웹서버 권한 탈취로 악성코드를 유포하는 경우가 많다. 이와 같이 악성코드 유포지는 웹서버를 악성코드 경유지로 삼아 신·변종의 악성코드를 PC에 대량으로 유포하는데 활용하는 URL 또는 IP주소를 의미한다. 바이러스, 웜 바이러스, 트로이목마 등을 사용자 PC에 유포해 네트워크 트래픽을 발생시키거나 시스템 성능저하, 파일삭제, 개인정보 유출, 원격제어 등의 심각한 피해를 입힐 수 있으며, DDoS 공격용 좀비 PC로도 감염시킬 수 있다.
이처럼 웹서버를 노린 공격이 기승을 부리면서 웹서버 보안의 중요성이 대두되고 있다. 이와 관련 웹서버 보안솔루션 전문업체 유엠브이기술의 조혁래 상무는 “웹서버 악성코드 공격을 통해 공격자는 웹페이지 소스코드를 열람하거나, 서버의 파일 및 DB 자료를 불법적으로 탈취한다”며 “악성코드 유포지 URL 또는 악성스크립트(iframe) 삽입을 통해 웹에 접속하는 고객 PC에 대량으로 악성코드를 유포시키거나 DDoS 공격을 유도할 수 있기 때문에 웹서버 보안에 각별한 주의를 기울여야 한다”고 강조했다.그렇다면 공격자는 왜 웹서버를 공격할까? 바로 다양한 정보유출이 가능하고 공격이 용이하기 때문이다. 공격자가 웹서버에 악성코드를 심고 실행시키면 DB서버를 장악할 수 있다. 이를 통해 다양한 개인정보 등이 저장돼 있는 DB정보를 탈취할 수 있다. 또한 웹서버 악성코드 파일 실행을 통해 홈페이지를 변조하거나 PC용 악성코드를 유포할 수도 있다. 이러한 악성코드는 시스템 명령어 네트워크 명령어 DB 접근 시스템 파일 접근 등의 유형으로 구성돼 다양한 공격이 가능하다 이와 관련 조 상무는 “시스템 명령어의 경우 시스템 정보를 열람하거나 시스템 Shutdown은 물론 특정 프로그램을 정지시키거나 Anti-virus 프로그램 등을 삭제할 수 있다”고 밝혔다. 덧붙여 그는 “네트워크 명령어의 경우 포트 스캐너, TELNET, SSH, FTP 등의 접속을 통해 내부 네트워크 시스템에 접근이 가능하다. 또한 DB 접근을 통해 데이터를 유출·변경·삭제하거나 시스템 파일 접근을 통해 키로그, 백도어 등의 업로드, 악성코드 삽입을 위한 파일 수정, 시스템 파일 삭제, 모든 시스템 디렉토리 열람 등이 가능해 웹서버 보안을 한층 강화해야 한다”고 당부했다. 그러므로 저희팀은 모두가 이용하지만 위에서와 같이 허점이 아직은 많은 웹서버를 주제로 어떻게 하면 보다 안전하게 이용을 할수있을까 생각을하다가 SEED암호와 웹서버를 합치면 어떨까라는 생각과 함께 졸업작품 을 시작하게 됬습니다.

1. 서론

1.1 연구의 필요성

미래에셋자산운용은 국내 금융사 최초로 웹서버 중 일부를 클라우드 형태로 운영하는 시스템을 도입했다고 지난 7일 밝혔다. 클라우드 서버 제공업체로는 세계적 전자상거래 기업인 아마

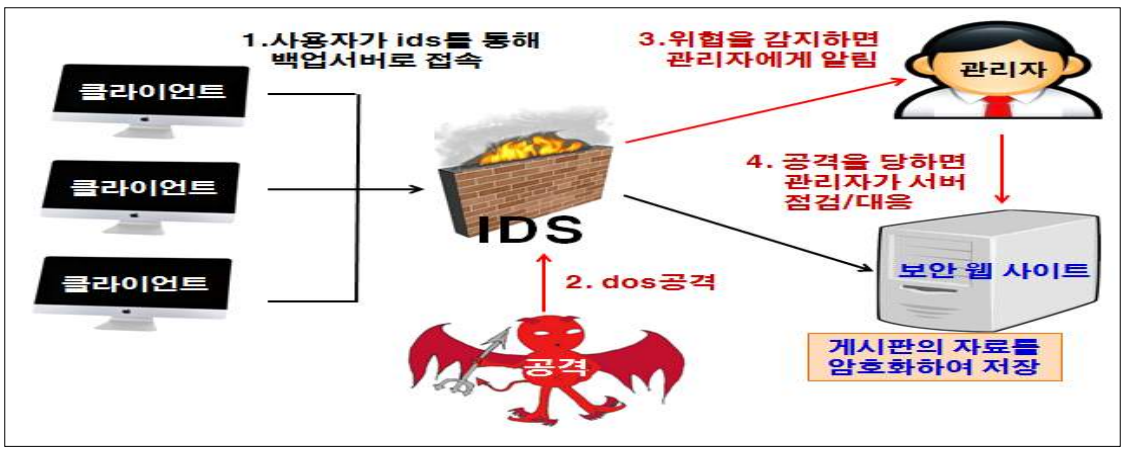
존의 계열사 '아마존웹서비스'를 선정했다. 미래에셋자산운용은 10여 개의 웹사이트를 통해 온라인 및 모바일 고객들과 소통해왔으며, 온라인 서비스의 중요성이 증가함에 따라 혁신적인 기술이라 평가 받는 클라우드 서비스를 도입하기로 결정했다.
클라우드형 웹 서버는 기업들이 자체적으로 보유하고 운영해오던 컴퓨팅 자원(서버 등)을 제3의 영역 즉, 클라우드 영역으로 옮겨두고 필요한 만큼의 자원을 사용하는 시스템이다. 이번에 클라우드형으로 적용하게 되는 웹서버는 그룹 글로벌홈페이지, 은퇴연구소홈페이지, 미래에셋자산운용 해외법인 홈페이지 등이며 공시나 기준가 산정 등과 연계된 국내웹사이트들은 추후 제도 개편 시 도입될 예정이다.
미래에셋은 서비스 도입 이후 방문 고객들이 현격히 개선된 접속 속도를 체감할 수 있을 것으로 기대하고 있다. 회사 자체 시뮬레이션 결과에 따르면 국내 접속자의 경우 약 50%, 해외 접속자의 경우 약 300% 접속 속도 개선 효과가 있는 것으로 나타났다. 또한 서버, 네트워크, 데이터베이스, 보안 등 웹 서비스 시설 요소들의 유지 관리가 일원화되어 연간 관리 비용이 절반 이상 절감된다는 것도 장점이다.
특히 글로벌 네트워크를 전세계 12개국까지 확장한 미래에셋자산운용의 경우 시카브(SICAV) 등 역외펀드의 우수한 투자 성과가 뒷받침 됨에 따라 해외 고객들의 웹 서비스 방문 빈도가 최근 활발히 증가하고 있다. 따라서 이미 글로벌 기업들에게는 활용도가 높은 클라우드 서비스 도입을 통해 해외 투자자의 눈높이에 맞는 웹 서비스 환경을 제공한다는 계획이다.
위와 같이 기업에서도 뜨고있는 클라우드 시스템을 웹서버에 적용시켜 글로벌 홈페이지를 작성하려고 하는 이때 암호를 이용한 보안 성능까지 갖춘다면 보다 안전하게 서버를 이용할수 있습니다.

1.2 과제 목표 및 내용

1. 과제 목표

PHP 에 대해서 파악 공부후 WEB서버를 구축합니다. 한국에서 사용가능한 오픈소스 암호에는 어떤 것들이 있는지 파악 후 이를 이용하여 PHP에 적용시킨다.
공격에 대비해 IDS를 설치하고 공격이 탐지가 되면 쉘 스크립트를 이용해 관리자에게 메일이 가도록 한다.

2. 구상도



<그림 1> 구상도

3. 추진 방법

SEED 암호를 사용하여 게시판 자료를 암호화 복호화 하고 Snort를 이용해 dos공격을 잡아낼것입니다. 웹서버는 DB를 구축해 회원을 관리하고 자기가 올린 게시글이 아닌 글은 SEED 암호를 이용해 암호화된 문자가 보이게 합니다.

2. 서버구축

2.1 PHP 소스

2.1.1 웹 서버

index.php

```
<html>
<head>
<title> Ghost Service 24 Hour </title>

<body>

<center><a href="gs24.php"><img src="./img/a5.png" vspace="200" hspace="250"
></a></center>

</body>
</html>
```

gs24.php

```
<?
        session_start();
?>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
<link rel="stylesheet" type="text/css" href="css/common.css">
</head>

<body>
<center>
<div id="wrap">
        <div id="header">
    <? include "./lib/top_login1.php"; ?>
        </div>  <!-- end of header -->

        <div id="menu">
        <? include "./lib/top_menu1.php"; ?>
        </div>  <!-- end of menu -->

  <div id="content">
        <div id="main_img"><img src="./img/main_img.jpg" width="1100" height="500"
></div>
  </div> <!-- end of content -->
</div> <!-- end of wrap -->
</center>
</body>
</html>
```

<그림 2> 메인화면

2.1.2 회원 가입

member_form.php

```
<?
        session_start();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta charset="euc-kr">
<link href="../css/common.css" rel="stylesheet" type="text/css" media="all">

<link href="../css/member.css" rel="stylesheet" type="text/css" media="all">
<script>
    function check_id()
    {

window.open("check_id.php?id="+document.member_form.id.value,"IDcheck","left=200,top=200,width=200,height=60,scrollbars=no,resizable=yes");
    }

    function check_nick()
    {

window.open("check_nick.php?nick="+document.member_form.nick.value,"NICKcheck","left=200,top=200,width=200,height=60,scrollbars=no,resizable=yes");
    }

    function check_input()
    {
        if (!document.member_form.id.value)

 {
            alert("아이디를 입력하세요");
            document.member_form.id.focus();
            return;

 }
```

```
        if (!document.member_form.pass.value)
        {
            alert("비밀번호를 입력하세요");
            document.member_form.pass.focus();
            return;
        }

        if (!document.member_form.pass_confirm.value)
        {
            alert("비밀번호확인을 입력하세요");
            document.member_form.pass_confirm.focus();
            return;
        }

        if (!document.member_form.name.value)
        {
            alert("이름을 입력하세요");
            document.member_form.name.focus();
            return;
        }

        if (!document.member_form.nick.value)
        {
            alert("닉네임을 입력하세요");
            document.member_form.nick.focus();

          return;
        }

        if (!document.member_form.hp2.value || !document.member_form.hp3.value )
        {
            alert("휴대폰 번호를 입력하세요");
            document.member_form.nick.focus();
            return;
        }

        if (document.member_form.pass.value !=
                document.member_form.pass_confirm.value)
        {

            alert("비밀번호가 일치하지 않습니다.\n다시 입력해주세요.");
            document.member_form.pass.focus();
            document.member_form.pass.select();
            return;
        }

      if                    (!document.member_form.email1.value                    ||
!document.member_form.email2.value )
        {
            alert("이메일 입력하세요");
            document.member_form.nick.focus();
            return;
        }

        document.member_form.submit();
    }

    function reset_form()

{
        document.member_form.id.value = "";
        document.member_form.pass.value = "";
        document.member_form.pass_confirm.value = "";
        document.member_form.name.value = "";
```

```
        document.member_form.nick.value = "";
        document.member_form.hp1.value = "010";
        document.member_form.hp2.value = "";
        document.member_form.hp3.value = "";
        document.member_form.email1.value = "";
        document.member_form.email2.value = "";

        document.member_form.id.focus();

        return;
    }
</script>
</head>

<body>
<div id="wrap">
  <div id="header">
    <? include "../lib/top_login2.php"; ?>
  </div>  <!-- end of header -->

  <div id="menu">
        <? include "../lib/top_menu2.php"; ?>
  </div>  <!-- end of menu -->

  <div id="content">
        <div id="col1">
        <div id="left_menu">

<?
        include "../lib/left_menu.php";
?>

        </div>
        </div> <!-- end of col1 -->

        <div id="col2">
        <form  name="member_form" method="post" action="insert.php">
        <div id="title">
        <img src="../img/title_join.gif">
        </div>

        <div id="form_join">
        <div id="join1">

        <ul>
        <li>* 아이디</li>
        <li>* 비밀번호</li>
        <li>* 비밀번호 확인</li>
        <li>* 이름</li>
        <li>* 닉네임</li>
        <li>* 휴대폰</li>
        <li>* 이메일</li>
        </ul>
        </div>
        <div id="join2">
        <ul>
        <li><div id="id1"><input type="text" name="id"></div><div id="id2"><a
href="#"><img src="../img/check_id.gif" onclick="check_id()"></a></div><div
id="id3">4~12자의 영문 소문자, 숫자와 특수기호(_) 만 사용할 수 있습니다.</div></li>

<li><input type="password" name="pass"></li>
        <li><input type="password" name="pass_confirm"></li>

<li><input type="text" name="name"></li>
```

```
<li><div id="nick1"><input type="text" name="nick"></div><div id="nick2" ><a

href="#"><img src="../img/check_id.gif" onclick="check_nick()"></a></div></li>
        <li><select class="hp" name="hp1">
              <option value='010'>010</option>
              <option value='011'>011</option>
              </select>    - <input type="text" class="hp" name="hp2"> - <input
type="text" class="hp" name="hp3"></li>
        <li><input type="text" id="email1" name="email1"> @ <input type="text"
name="email2"></li>
        </ul>
        </div>
        <div class="clear"></div>
        <div id="must"> * 는 필수 입력항목입니다.</div>
        </div>

        <div id="button"><a href="#"><img src="../img/button_save.gif"
onclick="check_input()"></a>  
                                    <a href="#"><img src="../img/button_reset.gif"
onclick="reset_form()"></a>
        </div>
            </form>
        </div> <!-- end of col2 -->
  </div> <!-- end of content -->
</div> <!-- end of wrap -->

</body>
</html>
```

insert.php

```
<meta charset="euc-kr">
<?
    $hp = $hp1."-".$hp2."-".$hp3;
    $email = $email1."@".$email2;

    $regist_day = date("Y-m-d (H:i)");  // 현재의 '년-월-일-시-분'을 저장
    $ip = $REMOTE_ADDR;           // 방문자의 IP 주소를 저장

    include "../lib/dbconn.php";         // dconn.php 파일을 불러옴

    $sql = "select * from member where id='$id'";
    $result = mysql_query($sql, $connect);
    $exist_id = mysql_num_rows($result);

    if($exist_id) {

        echo("
            <script>
              window.alert('해당 아이디가 존재합니다.')
              history.go(-1)
            </script>
          ");
          exit;
    }
    else
    {              // 레코드 삽입 명령을 $sql에 입력
              $sql = "insert into member(id, pass, name, nick, hp, email, regist_day,
level) ";
        $sql .= "values('$id', '$pass', '$name', '$nick', '$hp', '$email', '$regist_day',


9)";
```

```
mysql_query($sql, $connect);  // $sql 에 저장된 명령 실행
    }

    mysql_close();                  // DB 연결 끊기
    echo "
            <script>
             location.href = '../gs24.php';
            </script>
        ";
?>
```

check_id.php

```
<meta charset="euc-kr">
<?
    if(!$id)
    {
        echo("아이디를 입력하세요.");
    }
    else
    {
        include "../lib/dbconn.php";

        $sql = "select * from member where id='$id' ";

        $result = mysql_query($sql, $connect);
        $num_record = mysql_num_rows($result);

        if ($num_record)
        {

          echo "아이디가 중복됩니다!<br>";
           echo "다른 아이디를 사용하세요.<br>";
        }
        else
        {
           echo "사용가능한 아이디입니다.";
        }

        mysql_close();
    }
?>
```

check_nick.php

```
<meta charset="euc-kr">
<?
    if(!$nick)
    {
        echo("닉네임을 입력하세요.");
    }
    else
    {
        include "../lib/dbconn.php";

        $sql = "select * from member where nick='$nick' ";

        $result = mysql_query($sql, $connect);
        $num_record = mysql_num_rows($result);

if ($num_record)
        {
           echo "닉네임이 중복됩니다.<br>";
```

```
echo "다른 닉네임을 사용하세요.<br>";
        }
        else

          {
           echo "사용가능한 닉네임입니다.";
        }

        mysql_close();
    }
?>
```

<그림 3> 회원가입 화면

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| gs24               |
+--------------------+
2 rows in set (0.00 sec)

mysql>
mysql> show tables;
+-----------------+
| Tables_in_gs24 |
+-----------------+
| download       |
| member         |
+-----------------+
2 rows in set (0.00 sec)

mysql> select * from member;
+-------+------+-------+-------+---------------+-----------------+---------------------+-------+
| id    | pass | name  | nick  | hp            | email           | regist_day          | level |
+-------+------+-------+-------+---------------+-----------------+---------------------+-------+
| test2 | qwer | test2 | test2 | 010-1111-1111 | test2@test.com  | 2015-03-29 (14:39) |     9 |
| test  | 1234 | test  | test  | 010-0000-0000 | test@test.com   | 2015-03-28 (16:52) |     9 |
+-------+------+-------+-------+---------------+-----------------+---------------------+-------+
2 rows in set (0.00 sec)
```

<그림 4> 회원가입 DB

2.1.3 로그인

login.php

```
<?
        session_start();
```

```
?>
<meta charset="euc-kr">
<?


// 이전화면에서 이름이 입력되지 않았으면 "이름을 입력하세요"
   // 메시지 출력
   if(!$id) {
      echo("
            <script>
              window.alert('아이디를 입력하세요.')
              history.go(-1)

          </script>
           ");
           exit;
   }

   if(!$pass) {
      echo("
            <script>
              window.alert('비밀번호를 입력하세요.')
              history.go(-1)
            </script>
          ");
          exit;
   }
         include "../lib/dbconn.php";
         $sql = "select * from member where id='$id'";
   $result = mysql_query($sql, $connect);

   $num_match = mysql_num_rows($result);
         if(!$num_match)
   {
      echo("
            <script>
              window.alert('등록되지 않은 아이디입니다.')
              history.go(-1)
            </script>
          ");
    }
    else
    {
         $row = mysql_fetch_array($result);

         $db_pass = $row[pass];

         if($pass != $db_pass)
         {
             echo("
                 <script>
                   window.alert('비밀번호가 틀립니다.')
                   history.go(-1)
                 </script>
             ");

             exit;
         }
         else
         {

             $userid = $row[id];
             $username = $row[name];
             $usernick = $row[nick];
```

```
            $userlevel = $row[level];
            $_SESSION['userid'] = $userid;
            $_SESSION['username'] = $username;
            $_SESSION['usernick'] = $usernick;
            $_SESSION['userlevel'] = $userlevel;

            echo("
                <script>
                  location.href = '../gs24.php';
                </script>
            ");
        }
    }
?>
```

modify.php

```
<?
        session_start();
?>
<meta charset="euc-kr">
<?
    $hp = $hp1."-".$hp2."-".$hp3;
    $email = $email1."@".$email2;


    $regist_day = date("Y-m-d (H:i)");  // 현재의 '년-월-일-시-분'을 저장

    include "../lib/dbconn.php";          // dconn.php 파일을 불러옴

    $sql = "update member set pass='$pass', name='$name' , ";
    $sql  .= "nick='$nick',  hp='$hp',  email='$email',  regist_day='$regist_day'  where
id='$userid'";

    mysql_query($sql, $connect);  // $sql 에 저장된 명령 실행

    mysql_close();                  // DB 연결 끊기
    echo "
            <script>
             location.href = '../gs24.php';
            </script>
        ";
?>
```

logout.php

```
<?
   session_start();
   unset($_SESSION['userid']);
   unset($_SESSION['username']);
   unset($_SESSION['usernick']);
   unset($_SESSION['userlevel']);

   echo("
         <script>
            location.href = '../gs24.php';
          </script>
        ");
?>
```

-- **login_form.php** --

```
<? session_start(); ?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
```

```
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta charset="euc-kr">
<link href="../css/common.css" rel="stylesheet" type="text/css" media="all">

<link href="../css/member.css" rel="stylesheet" type="text/css" media="all">
</head>

<body>
<div id="wrap">
  <div id="header">
    <? include "../lib/top_login2.php"; ?>
  </div>  <!-- end of header -->

  <div id="menu">
        <? include "../lib/top_menu2.php"; ?>
  </div>  <!-- end of menu -->


        <div id="content">
        <div id="col1">
        <div id="left_menu">
<?
        include "../lib/left_menu.php";
?>
        </div>
        </div> <!-- end of col1 -->

        <div id="col2">

        <form  name="member_form" method="post" action="login.php">
        <div id="title">
        <img src="../img/title_login.gif">
        </div>

        <div id="login_form">

         <div class="clear"></div>


         <div id="login2">
        <div id="id_input_button">
        <div id="id_pw_title">
        <ul>
        <li><img src="../img/id_title.gif"></li>
                <li><img src="../img/pw_title.gif"></li>
        </ul>
        </div>
        <div id="id_pw_input">
        <ul>
        <li><input type="text" name="id" class="login_input"></li>
        <li><input type="password" name="pass" class="login_input"></li>
        </ul>
        </div>
        <div id="login_button">
        <input type="image" src="../img/login_button.gif">
        </div>
        </div>
        <div class="clear"></div>

        <div id="login_line"></div>
        <div                                                    id="join_button"><img
src="../img/no_join.gif">    <a
```

```
href="../member/member_form.php"><img src="../img/join_button.gif"></a></div>

</div>
        </div> <!-- end of form_login -->

            </form>

</div> <!-- end of col2 -->
  </div> <!-- end of content -->
</div> <!-- end of wrap -->

</body>
</html>
```

member_form_modify.php

```
<?
    session_start();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<meta charset="euc-kr">
<link href="../css/common.css" rel="stylesheet" type="text/css" media="all">
<link href="../css/member.css" rel="stylesheet" type="text/css" media="all">
<script>
   function check_id()
   {
      window.open("check_id.php?id=" + document.member_form.id.value,
           "IDcheck",
            "left=200,top=200,width=250,height=100,scrollbars=no,resizable=yes");

         }

   function check_nick()
   {
      window.open("check_nick.php?nick=" + document.member_form.nick.value,
           "NICKcheck",
            "left=200,top=200,width=250,height=100,scrollbars=no,resizable=yes");
   }

   function check_input()
   {
       if (!document.member_form.pass.value)
       {
            alert("비밀번호를 입력하세요");
            document.member_form.pass.focus();
            return;
       }

       if (!document.member_form.pass_confirm.value)
       {
            alert("비밀번호확인을 입력하세요");
            document.member_form.pass_confirm.focus();
            return;
       }

       if (!document.member_form.name.value)
       {
            alert("이름을 입력하세요");
            document.member_form.name.focus();
            return;
```

```
 }


          if (!document.member_form.nick.value)
       {
           alert("닉네임을 입력하세요");
           document.member_form.nick.focus();

         return;
       }

       if (!document.member_form.hp2.value || !document.member_form.hp3.value )
       {
           alert("휴대폰 번호를 입력하세요");
           document.member_form.nick.focus();
           return;
       }

       if (document.member_form.pass.value !=
             document.member_form.pass_confirm.value)
       {
           alert("비밀번호가 일치하지 않습니다.\n다시 입력해주세요.");
           document.member_form.pass.focus();
           document.member_form.pass.select();

         return;
       }
if (!document.member_form.email1.value || !document.member_form.email2.value )
       {
           alert("이메일 입력하세요");
           document.member_form.nick.focus();
           return;
       }

       document.member_form.submit();
   }


         function reset_form()
   {
       document.member_form.id.value = "";
       document.member_form.pass.value = "";
       document.member_form.pass_confirm.value = "";
       document.member_form.name.value = "";
       document.member_form.nick.value = "";
       document.member_form.hp1.value = "010";
       document.member_form.hp2.value = "";
       document.member_form.hp3.value = "";
       document.member_form.email1.value = "";
       document.member_form.email2.value = "";

       document.member_form.id.focus();

       return;
   }
</script>
</head>
<?
    include "../lib/dbconn.php";

    $sql = "select * from member where id='$userid'";
    $result = mysql_query($sql, $connect);

    $row = mysql_fetch_array($result);
```

```
    $hp = explode("-", $row[hp]);
    $hp1 = $hp[0];
    $hp2 = $hp[1];

$hp3 = $hp[2];

    $email = explode("@", $row[email]);

        $email1 = $email[0];
    $email2 = $email[1];

    mysql_close();
?>
<body>
<div id="wrap">
  <div id="header">
    <? include "../lib/top_login2.php"; ?>
  </div>  <!-- end of header -->

  <div id="menu">
        <? include "../lib/top_menu2.php"; ?>
  </div>  <!-- end of menu -->

  <div id="content">
        <div id="col1">
        <div id="left_menu">

<?
        include "../lib/left_menu.php";
?>
        </div>
        </div> <!-- end of col1 -->

        <div id="col2">
        <form  name="member_form" method="post" action="modify.php">
        <div id="title">
        <img src="../img/title_member_modify.gif">
        </div>



        <div id="form_join">
        <div id="join1">
        <ul>
        <li>* 아이디</li>
        <li>* 비밀번호</li>
        <li>* 비밀번호 확인</li>
        <li>* 이름</li>
        <li>* 닉네임</li>
        <li>* 휴대폰</li>
        <li>* 이메일</li>
        </ul>
        </div>
        <div id="join2">
        <ul>
        <li><?= $row[id] ?></li>
        <li><input type="password" name="pass" value="<?= $row[pass] ?>"></li>
        <li><input  type="password"  name="pass_confirm"  value="<?=  $row[pass]
?>"></li>
        <li><input type="text" name="name" value="<?= $row[name] ?>"></li>
        <li><div  id="nick1"><input  type="text"  name="nick"  value="<?=  $row[nick]
?>"></div><div  id="nick2"  ><a  href="#"><img  src="../img/check_id.gif"
onclick="check_nick()"></a></div></li>
        <li><input type="text" class="hp" name="hp1" value="<?= $hp1 ?>">
```

```
 - <input type="text" class="hp" name="hp2" value="<?= $hp2 ?>"> - <input type="text" class="hp" name="hp3" value="<?= $hp3 ?>"></li>
        <li><input type="text" id="email1" name="email1" value="<?= $email1 ?>"> @ <input type="text" name="email2"
                    value="<?= $email2 ?>"></li>

        </ul>
        </div>

        <div class="clear"></div>
        <div id="must"> * 는 필수 입력항목입니다.^^</div>
        </div>

        <div id="button"><a href="#"><img src="../img/button_save.gif" onclick="check_input()"></a>  

<a href="#"><img src="../img/button_reset.gif" onclick="reset_form()"></a>

        </div>
            </form>

        </div>
  </div>
</div>


</body>
</html>
```

<그림 5> 로그인 화면

2.1.4 게시판

view.php

```
<?
        session_start();
        include "../lib/dbconn.php";
```

```
$sql = "select * from $table where num=$num";


$result = mysql_query($sql, $connect);

    $row = mysql_fetch_array($result);

        $item_num       = $row[num];


        $item_id        = $row[id];
        $item_name      = $row[name];
        $item_nick      = $row[nick];
        $item_hit       = $row[hit];

        $file_name[0]    = $row[file_name_0];
        $file_name[1]    = $row[file_name_1];
        $file_name[2]    = $row[file_name_2];

        $file_type[0]    = $row[file_type_0];
        $file_type[1]    = $row[file_type_1];
        $file_type[2]    = $row[file_type_2];

        $file_copied[0] = $row[file_copied_0];
        $file_copied[1] = $row[file_copied_1];
        $file_copied[2] = $row[file_copied_2];

    $item_date      = $row[regist_day];
        $item_subject = str_replace(" ", " ", $row[subject]);

        $item_content = str_replace(" ", " ", $row[content]);
        $item_content = str_replace("\n", "<br>", $item_content);
        $new_hit = $item_hit + 1;

        $sql = "update $table set hit=$new_hit where num=$num";   // 글 조회수
증가시킴
        mysql_query($sql, $connect);
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta charset="euc-kr">
<link href="../css/common.css" rel="stylesheet" type="text/css" media="all">
<link href="../css/board3.css" rel="stylesheet" type="text/css" media="all">
<script>
    function del(href)
    {
        if(confirm("한번 삭제한 자료는 복구할 방법이 없습니다.\n\n정말
삭제하시겠습니까?")) {
                document.location.href = href;
        }
    }
</script>
</head>

<body>
<div id="wrap">


<div id="header">
    <? include "../lib/top_login2.php"; ?>
  </div>  <!-- end of header -->
```

```
<div id="menu">
        <? include "../lib/top_menu2.php"; ?>
  </div>  <!-- end of menu -->


<div id="content">
        <div id="col1">
        <div id="left_menu">
<?
        include "../lib/left_menu.php";

        ?>

        </div>
        </div> <!-- end of col1 -->

        <div id="col2">
        <div id="title">
        <img src="../img/title_download.gif">
        </div>
        <div id="view_comment">  </div>

        <div id="view_title">
        <div id="view_title1"><?= $item_subject ?></div><div id="view_title2"><?=
$item_nick ?> | 조회 : <?= $item_hit ?>
                                | <?= $item_date ?> </div>
        </div>

        <div id="view_content">
<?
        for ($i=0; $i<3; $i++)
        {
        if ($userid && $file_copied[$i])
        {
        $show_name = $file_name[$i];
        $real_name = $file_copied[$i];
        $real_type = $file_type[$i];
        $file_path = "./data/".$real_name;
        $file_size = filesize($file_path);

        echo "▷ 첨부파일 : $show_name ($file_size Byte)     
                <a
href='download.php?table=$table&num=$num&real_name=$real_name&show_name=$s
how_name&file_type=$real_type'>[저장]</a><br>";
        }
        }
?>
            <br>
<?
        if($userid && $userid==$item_id)
        {
?>
        <?= $item_content ?>
<?
        }
?>

<?
   include 'class.crypto.php';
   $crypto = new Crypto();

if($userid && $userid!=$item_id)

        {
```

```
?>
        <?= $crypto->encrypt($item_content) ?>
<?
        }
?>
        </div>

        <div id="view_button">
        <a href="list.php?table=<?=$table?>&page=<?=$page?>"><img
src="../img/list.png"></a> 
<?

        if($userid && $userid==$item_id)
        {
?>

        <a
href="write_form.php?table=<?=$table?>&mode=modify&num=<?=$num?>&page=<?=$p
age?>"><img src="../img/modify.png"></a> 
        <a
href="javascript:del('delete.php?table=<?=$table?>&num=<?=$num?>')"><img
src="../img/delete.png"></a> 
<?
        }
?>
<?
        if($userid)
        {
?>
        <a href="write_form.php?table=<?=$table?>"><img src="../img/write.png"></a>
<?
        }
?>
        </div>
        <div class="clear"></div>

        </div> <!-- end of col2 -->
  </div> <!-- end of content -->
</div> <!-- end of wrap -->

</body>
</html>
```

insert.php

```
<? session_start(); ?>

<meta charset="euc-kr">
<?

    if(!$userid) {
        echo("
        <script>
          window.alert('로그인 후 이용해 주세요.')
          history.go(-1)
        </script>
        ");
        exit;
    }
    $regist_day = date("Y-m-d (H:i)");  // 현재의 '년-월-일-시-분'을 저장

    // 다중 파일 업로드
    $files = $_FILES["upfile"];
```

```
    $count = count($files["name"]);
    $upload_dir = './data/';

    for ($i=0; $i<$count; $i++)
    {
        $upfile_name[$i]      = $files["name"][$i];


$upfile_tmp_name[$i] = $files["tmp_name"][$i];
        $upfile_type[$i]      = $files["type"][$i];


$upfile_size[$i]      = $files["size"][$i];
        $upfile_error[$i]     = $files["error"][$i];

        $file = explode(".", $upfile_name[$i]);
        $file_name = $file[0];
        $file_ext  = $file[1];



          If (!$upfile_error[$i])
        {
            $new_file_name = date("Y_m_d_H_i_s");
            $new_file_name = $new_file_name."_".$i;
            $copied_file_name[$i] = $new_file_name.".".$file_ext;
            $uploaded_file[$i] = $upload_dir.$copied_file_name[$i];

            if( $upfile_size[$i]  > 5000000 ) {
                echo("
                <script>
                alert('업로드 파일 크기가 지정된 용량(5MB)을 초과합니다!<br>파일 크기를
체크해주세요! ');
                history.go(-1)
                </script>
                ");
                exit;
            }

            if (!move_uploaded_file($upfile_tmp_name[$i], $uploaded_file[$i]) )
            {
                echo("
                    <script>
                    alert('파일을 지정한 디렉토리에 복사하는데 실패했습니다.');
                    history.go(-1)
                    </script>
                ");
                exit;
            }
        }
    }


    include "../lib/dbconn.php";          // dconn.php 파일을 불러옴
    if ($mode=="modify")
    {
        $num_checked = count($_POST['del_file']);
        $position = $_POST['del_file'];

        for($i=0; $i<$num_checked; $i++)                    // delete checked item
        {
            $index = $position[$i];
            $del_ok[$index] = "y";
        }
```

```
        $sql = "select * from $table where num=$num";    // get target record
        $result = mysql_query($sql);
        $row = mysql_fetch_array($result);

        for ($i=0; $i<$count; $i++)                  // update DB with the value of file
input box
        {
            $field_org_name = "file_name_".$i;
            $field_real_name = "file_copied_".$i;

            $org_name_value = $upfile_name[$i];

            $org_real_value = $copied_file_name[$i];
            if ($del_ok[$i] == "y")
            {

                $delete_field = "file_copied_".$i;

                $delete_name = $row[$delete_field];
                $delete_path = "./data/".$delete_name;
                unlink($delete_path);

                $sql = "update $table set $field_org_name = '$org_name_value',

                    $field_real_name = '$org_real_value'   where num=$num";

            mysql_query($sql, $connect);  // $sql 에 저장된 명령 실행
            }
            else
            {
                if (!$upfile_error[$i])
                {
                    $sql = "update $table set $field_org_name = '$org_name_value',
$field_real_name = '$org_real_value'   where num=$num";
                    mysql_query($sql, $connect);  // $sql 에 저장된 명령 실행
                }
            }

        }


$sql = "update $table set subject='$subject', content='$content' where num=$num";
        mysql_query($sql, $connect);  // $sql 에 저장된 명령 실행
    }

          else
    {

      $sql = "insert into $table (id, name, nick, subject, content, regist_day, hit, ";

      $sql .= " file_name_0, file_name_1, file_name_2, file_type_0, file_type_1,
file_type_2, file_copied_0,   file_copied_1, file_copied_2) ";

      $sql .= " values('$userid', '$username', '$usernick', '$subject', '$content',
'$regist_day', 0, ";

      $sql .= " '$upfile_name[0]', '$upfile_name[1]',   '$upfile_name[2]',
'$upfile_type[0]', '$upfile_type[1]',   '$upfile_type[2]', ";

       $sql .= " '$copied_file_name[0]', '$copied_file_name[1]','$copied_file_name[2]')";

            mysql_query($sql, $connect);  // $sql 에 저장된 명령 실행
    }
```

```
mysql_close();                  // DB 연결 끊기

   echo "

      <script>

        location.href = 'list.php?table=$table&page=$page';
      </script>
   ";
?>
```

list.php

```
<?
        session_start();
        $table = "download";
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta charset="euc-kr">
<link href="../css/common.css" rel="stylesheet" type="text/css" media="all">
<link href="../css/board3.css" rel="stylesheet" type="text/css" media="all">
</head>
<?

        include "../lib/dbconn.php";
$scale=10;      // 한 화면에 표시되는 글 수

    if ($mode=="search")
        {
        if(!$search)
        {

        echo("
        <script>
         window.alert('검색할 단어를 입력해 주세요!');
              history.go(-1);
        </script>
        ");
        exit;
        }

        $sql = "select * from $table where $find like '%$search%' order by num
desc";
        }
        else
        {
        $sql = "select * from $table order by num desc";
        }

        $result = mysql_query($sql, $connect);
        $total_record = mysql_num_rows($result); // 전체 글 수

        // 전체 페이지 수($total_page) 계산
        if ($total_record % $scale == 0)
        $total_page = floor($total_record/$scale);
        else
        $total_page = floor($total_record/$scale) + 1;

        if (!$page)                 // 페이지번호($page)가 0 일 때
        $page = 1;              // 페이지 번호를 1로 초기화
```

```
// 표시할 페이지($page)에 따라 $start 계산
        $start = ($page - 1) * $scale;
        $number = $total_record - $start;
?>
<body>
<div id="wrap">
  <div id="header">
    <? include "../lib/top_login2.php"; ?>
  </div>  <!-- end of header -->

  <div id="menu">


<? include "../lib/top_menu2.php"; ?>
  </div>  <!-- end of menu -->

  <div id="content">
        <div id="col1">
        <div id="left_menu">
<?
        include "../lib/left_menu.php";
?>
        </div>
        </div>

        <div id="col2">
        <div id="title">
        <img src="../img/title_download.gif">
        </div>

        <form  name="board_form" method="post"
action="list.php?table=<?=$table?>&mode=search">
        <div id="list_search">
        <div id="list_search1">▷ 총 <?= $total_record ?> 개의 게시물이 있습니다.
</div>
        <div id="list_search2"><img src="../img/select_search.gif"></div>
        <div id="list_search3">

        <select name="find">
                <option value='subject'>제목</option>
                <option value='content'>내용</option>
                <option value='nick'>별명</option>
                <option value='name'>이름</option>
        </select></div>
        <div id="list_search4"><input type="text" name="search"></div>
        <div id="list_search5"><input type="image"
src="../img/list_search_button.gif"></div>
        </div>
        </form>

        <div class="clear"></div>
        <div id="list_top_title">
        <ul>
        <li id="list_title1"><img src="../img/list_title1.gif"></li>
        <li id="list_title2"><img src="../img/list_title2.gif"></li>
        <li id="list_title3"><img src="../img/list_title3.gif"></li>
        <li id="list_title4"><img src="../img/list_title4.gif"></li>
        <li id="list_title5"><img src="../img/list_title5.gif"></li>
        </ul>
        </div>

        <div id="list_content">
<?
   for ($i=$start; $i<$start+$scale && $i < $total_record; $i++)
```

```
        {
      mysql_data_seek($result, $i);
      // 가져올 레코드로 위치(포인터) 이동

        $row = mysql_fetch_array($result);
      // 하나의 레코드 가져오기

          $item_num      = $row[num];
          $item_id       = $row[id];
          $item_name     = $row[name];
          $item_nick     = $row[nick];
          $item_hit      = $row[hit];

      $item_date     = $row[regist_day];
          $item_date = substr($item_date, 0, 10);

          $item_subject = str_replace(" ", " ", $row[subject]);
?>
        <div id="list_item">
        <div id="list_item1"><?= $number ?></div>
        <div id="list_item2"><a
href="view.php?table=<?=$table?>&num=<?=$item_num?>&page=<?=$page?>"><?=
$item_subject ?></a></div>
        <div id="list_item3"><?= $item_nick ?></div>
        <div id="list_item4"><?= $item_date ?></div>
        <div id="list_item5"><?= $item_hit ?></div>
        </div>
<?
          $number--;
   }
?>
        <div id="page_button">
        <div id="page_num"> ◀ 이전     
<?
   // 게시판 목록 하단에 페이지 링크 번호 출력
   for ($i=1; $i<=$total_page; $i++)
   {
        if ($page == $i)      // 현재 페이지 번호 링크 안함

        {
        echo "<b> $i </b>";
        }
        else
        {
        echo "<a href='list.php?table=$table&page=$i'> $i </a>";
        }
   }
?>
            다음 ▶
        </div>
        <div id="button">
        <a href="list.php?table=<?=$table?>&page=<?=$page?>"><img
src="../img/list.png"></a> 
<?
        if($userid)
        {
?>
        <a href="write_form.php?table=<?=$table?>"><img src="../img/write.png"></a>
<?
        }
?>
        </div>
        </div> <!-- end of page_button -->
```

```
</div> <!-- end of list content -->

        <div class="clear"></div>

        </div> <!-- end of col2 -->

  </div> <!-- end of content -->
</div> <!-- end of wrap -->

</body>

</html>
```

download.php

```
<? session_start(); ?>
<?

        if(!$userid) {

        echo("

        <script>

              window.alert('로그인 후 이용해 주세요.')
              history.go(-1)
            </script>
        ");

        exit;
        }

    $file_path = "./data/".$real_name;

    if( file_exists($file_path) )
    {
            $fp = fopen($file_path,"rb");

            if( $file_type )
            {
        Header("Content-type: application/x-msdownload");
                Header("Content-Length: ".filesize($file_path));
                Header("Content-Disposition: attachment; filename=$show_name");
                Header("Content-Transfer-Encoding: binary");
        Header("Content-Description: File Transfer");

                header("Expires: 0");

        }
             else
             {
                if(eregi("(MSIE 5.0|MSIE 5.1|MSIE 5.5|MSIE 6.0)",
$HTTP_USER_AGENT))
                {
                     Header("Content-type: application/octet-stream");
                     Header("Content-Length: ".filesize($file_path));
                     Header("Content-Disposition: attachment;
filename=$show_name");
                     Header("Content-Transfer-Encoding: binary");
                     Header("Expires: 0");
                 }
                 else
```

```
{
                        Header("Content-type: file/unknown");
                        Header("Content-Length: ".filesize($file_path));
                        Header("Content-Disposition: attachment;
filename=$show_name");
                        Header("Content-Description: PHP3 Generated Data");
                        Header("Expires: 0");
                    }
                }

          if(!fpassthru($fp))
          fclose($fp);
          }
?>
```

delete.php

```
<? session_start(); ?>
<?
          if(!$userid) {
          echo("
          <script>
                window.alert('로그인 후 이용해 주세요.')
                history.go(-1)
            </script>
          ");
          exit;
          }
     $file_path = "./data/".$real_name;

     if( file_exists($file_path) )
     {
               $fp = fopen($file_path,"rb");

               if( $file_type )
               {
          Header("Content-type: application/x-msdownload");
                    Header("Content-Length: ".filesize($file_path));
                    Header("Content-Disposition: attachment; filename=$show_name");
                    Header("Content-Transfer-Encoding: binary");
          Header("Content-Description: File Transfer");

                    header("Expires: 0");
               }
               else
               {


 if(eregi("(MSIE 5.0|MSIE 5.1|MSIE 5.5|MSIE 6.0)", $HTTP_USER_AGENT))
                    {
                         Header("Content-type: application/octet-stream");

                    Header("Content-Length: ".filesize($file_path));
                         Header("Content-Disposition: attachment;
filename=$show_name");

Header("Content-Transfer-Encoding: binary");
                         Header("Expires: 0");
                    }

                              else

                    {
```

```
    Header("Content-type: file/unknown");
                        Header("Content-Length: ".filesize($file_path));
                        Header("Content-Disposition: attachment;
filename=$show_name");

                        Header("Content-Description: PHP3 Generated Data");
                        Header("Expires: 0");
                    }
                }

          if(!fpassthru($fp))

          fclose($fp);
          }
?
>
```

write_form.php

```
<?
          session_start();
          include "../lib/dbconn.php";

          if ($mode=="modify")
          {
          $sql = "select * from $table where num=$num";
          $result = mysql_query($sql, $connect);

          $row = mysql_fetch_array($result);

          $item_subject       = $row[subject];
          $item_content       = $row[content];

          $item_file_0 = $row[file_name_0];
          $item_file_1 = $row[file_name_1];
          $item_file_2 = $row[file_name_2];

          $copied_file_0 = $row[file_copied_0];
          $copied_file_1 = $row[file_copied_1];
          $copied_file_2 = $row[file_copied_2];
          }
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta charset="euc-kr">
<link href="../css/common.css" rel="stylesheet" type="text/css" media="all">
<link href="../css/board3.css" rel="stylesheet" type="text/css" media="all">
<script>
  function check_input()
   {
       if (!document.board_form.subject.value)
       {
           alert("제목을 입력하세요1");
           document.board_form.subject.focus();
           return;
       }

       if (!document.board_form.content.value)

         {
           alert("내용을 입력하세요!");
```

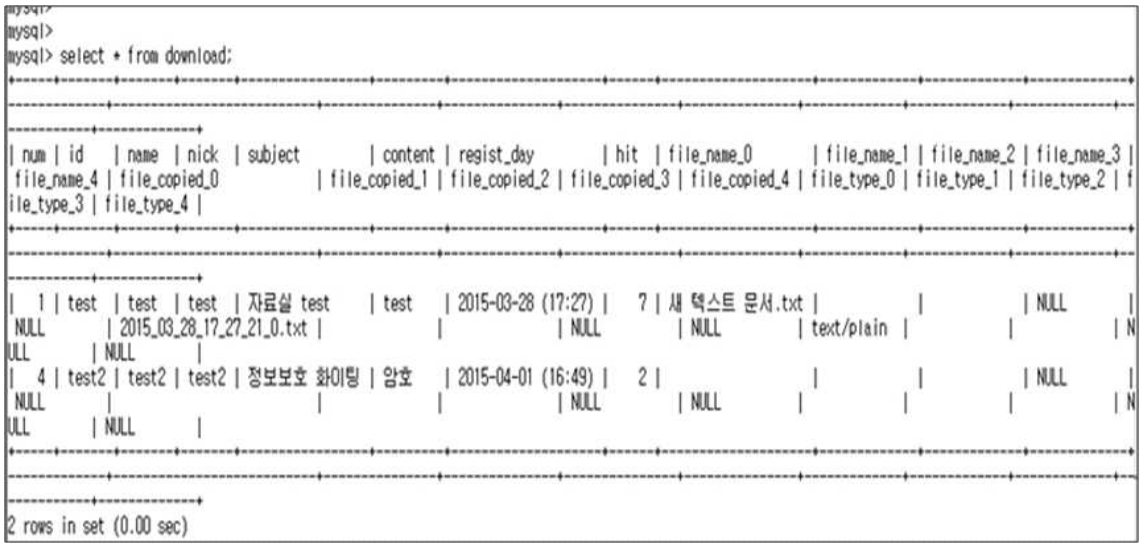
```
document.board_form.content.focus();
            return;
        }
        document.board_form.submit();
    }
</script>
</head>

<body>
<div id="wrap">

  <div id="header">
    <? include "../lib/top_login2.php"; ?>
  </div>  <!-- end of header -->

  <div id="menu">
        <? include "../lib/top_menu2.php"; ?>
  </div>  <!-- end of menu -->


<div id="content">
        <div id="col1">
        <div id="left_menu">
<?
        include "../lib/left_menu.php";
?>
        </div>
        </div> <!-- end of col1 -->

        <div id="col2">
        <div id="title">
        <img src="../img/title_download.gif">
        </div>
        <div class="clear"></div>

        <div id="write_form_title">
        <img src="../img/write_form_title.gif">
        </div>

        <div class="clear"></div>
<?
        if($mode=="modify")
        {

?>
        <form  name="board_form" method="post"
action="insert.php?mode=modify&num=<?=$num?>&page=<?=$page?>&table=<?=$table
?>" enctype="multipart/form-data">
<?
        }
        else
        {
?>
        <form  name="board_form" method="post"
action="insert.php?table=<?=$table?>" enctype="multipart/form-data">
<?
        }
?>
        <div id="write_form">
        <div class="write_line"></div>
        <div id="write_row1"><div class="col1"> 닉네임 </div><div
class="col2"><?=$usernick?></div></div>
        <div class="write_line"></div>
```

```
<div id="write_row2"><div class="col1"> 제목    </div>
                                <div class="col2"><input type="text" name="subject"
value="<?=$item_subject?>" ></div>
        </div>
                <div class="write_line"></div>
        <div id="write_row3"><div class="col1"> 내용    </div>
                                <div class="col2"><textarea rows="15" cols="79"
name="content"><?=$item_content?></textarea></div>
        </div>
        <div class="write_line"></div>
                <div id="write_row4"><div class="col1"> 첨부파일1    </div>
                                <div class="col2"><input type="file" name="upfile[]"> *
5MB까지 업로드 가능!</div>
        </div>
        <div class="clear"></div>
<?      if ($mode=="modify" && $item_file_0)
        {
?>
        <div class="delete_ok"><?=$item_file_0?> 파일이 등록되어 있습니다. <input
type="checkbox" name="del_file[]" value="0"> 삭제</div>

        <div class="clear"></div>

<?
        }
?>

<div class="write_line"></div>

<div id="write_row5"><div class="col1"> 첨부파일2  </div>
                                <div class="col2"><input type="file" name="upfile[]">  *
5MB까지 업로드 가능!</div>
        </div>
<?      if ($mode=="modify" && $item_file_1)
        {
?>

<div class="delete_ok"><?=$item_file_1?> 파일이 등록되어 있습니다. <input
type="checkbox" name="del_file[]" value="1"> 삭제</div>

<div class="clear"></div>
<?
        }
?>
        <div class="write_line"></div>
        <div class="clear"></div>
        <div id="write_row6"><div class="col1"> 첨부파일3    </div>
                                <div class="col2"><input type="file" name="upfile[]">  *
5MB까지 업로드 가능!</div>
        </div>

<?      if ($mode=="modify" && $item_file_2)
        {
?>

<div class="delete_ok"><?=$item_file_2?> 파일이 등록되어 있습니다. <input
type="checkbox" name="del_file[]" value="2"> 삭제</div>

<div class="clear"></div>

<?
        }
?>
        <div class="write_line"></div>
```

```
        <div class="clear"></div>
        </div>

        <div id="write_button"><a href="#"><img src="../img/ok.png"
onclick="check_input()"></a> 

        <a href="list.php?table=<?=$table?>&page=<?=$page?>"><img
src="../img/list.png"></a>
        </div>


        </form>

        </div> <!-- end of col2 -->
  </div> <!-- end of content -->
</div> <!-- end of wrap -->

</body>
</html>
```

<그림 6> 자료실 화면

<그림 7> 글쓰기



```
mysql>
mysql> select * from download;
+-----+-------+-------+-------+-----------------+---------+---------------------+-----+---------------------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+
| num | id    | name  | nick  | subject         | content | regist_day          | hit | file_name_0         | file_name_1 | file_name_2 | file_name_3 | file_name_4 | file_copied_0           | file_copied_1 | file_copied_2 | file_copied_3 | file_copied_4 | file_type_0 | file_type_1 | file_type_2 | file_type_3 | file_type_4 |
+-----+-------+-------+-------+-----------------+---------+---------------------+-----+---------------------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+
|   1 | test  | test  | test  | 자료실 test     | test    | 2015-03-28 (17:27) |   7 | 새 텍스트 문서.txt |           |           | NULL      | NULL      | 2015_03_28_17_27_21_0.txt |           |           | NULL      | NULL      | text/plain  |           |           | NULL        | NULL        |
|   4 | test2 | test2 | test2 | 정보보호 화이팅 | 암호    | 2015-04-01 (16:49) |   2 |                     |           |           | NULL      | NULL      |           |           |           | NULL      | NULL      |           |           |           | NULL        | NULL        |
+-----+-------+-------+-------+-----------------+---------+---------------------+-----+---------------------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+
2 rows in set (0.00 sec)
```

<그림 8> 자료실 DB

3.SEED암호

3.1 SEED암호

3.1.1 SEED 암호란?

SEED는 1999년 2월 한국정보보호진흥원(한국인터넷진흥원의 전신)의 기술진이 개발한 128비트 및 256비트 대칭 키 블록 암호 알고리즘으로, 미국에서 수출되는 웹브라우저 보안 수준이 40비트로 제한됨에 따라 128비트 보안을 위해 별도로 개발된 알고리즘이다.

대한민국의 인터넷 뱅킹 및 쇼핑 환경에서 자주 쓰이나, SEED 기반 보안 프로그램은 액티브엑스 플러그인으로 배포될 수밖에 없기 때문에 공인인증서와 함께 대한민국의 웹 호환성 문제와 가장 밀접하게 연관되어 있다. 국내 보안업체들은 128비트 보안을 위한 별도 프로그램을 개발하기 위해 노력했으며, 그렇게 개발된 프로그램이 웹브라우저 플러그인으로 배포되어 국내 인터넷 뱅킹에 사용되기 시작했다. 2000년까지 미국 정부는 미국 내에서 사용되는 웹브라우저에서는 128비트 수준의 보안 접속을 허용했으나, 미국 외로 수출되는 인터넷 익스플로러나 넷스케이프 등의 웹브라우저에서는 40비트 이상의 보안 접속을 금지했다. 40비트 수준의 암호화는 1997년경의 연산능력으로도 3.5시간만에 깨지는 무용지물이었다.

3.1.2 SEED 암호 알고리즘이란?

SEED 암호화 알고리즘은 민간 부분인 인터넷, 전자상거래, 무선 통신 등에서 공개 시 민감한 영향을 미칠 수 있는 정보의 보호황 개인 프라이버시 등을 보호하기 위하여 1999년 2월 한국정보보호센터(KISA)에 의해 개발 된 블록암호알고리즘이다. 대칭키 블록 암호알고리즘으로 비밀성을 제공하는 암호시스템의 중요 요소이다. n비트 블록 암호알고리즘이란 고정 된 n비트 평문을 같은 길이의 n 비트 암호문으로 바꾸는 함수를 말한다.

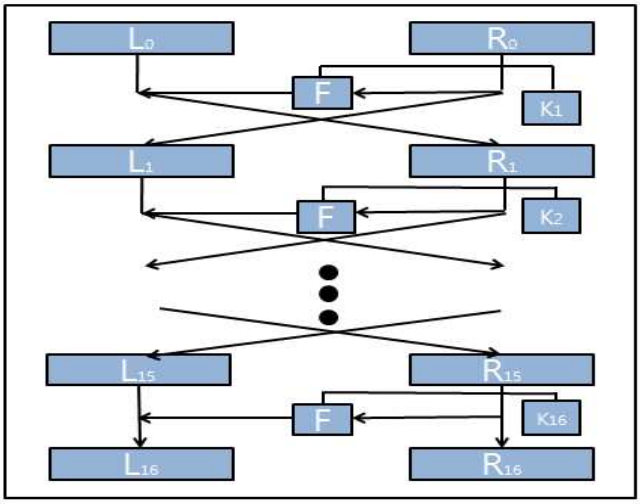
이러한 변형 과정에 암/복호키가 적용하여 암호화와 복호화를 수행한다.

Feistel 구조란 각각 n/2 비트인 L_0, R_0 블록으로 이루어진 n 비트 평문 블록 (L_0, R_0)이 r 라

운드(r≥1)를 거쳐 암호문(L_r,R_r)으로 반환되는 반복 구조이다.

3.1.3 SEED 알고리즘 구조

SEED 알고리즘의 전체 구조는 Feistel 구조로 이루어져 있으며, 128비트의 평문 블록과 128비트 키를 입력으로 사용하여 총 16라운드를 거쳐 128비트 암호문 블록을 출력한다.



- Feistel 구조
- 128비트 암호문 생성
- F함수는 K_i 에서 키값을 받아 64비트 블록 출력
- G함수를 이용한 라운드 키(K_i)생성
- G함수는 S-BOX를 이용

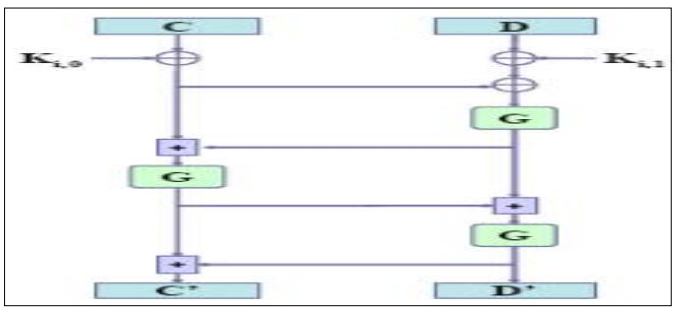
<그림 9> SEED알고리즘

3.1.4 F함수 & G함수

F함수:Feistel 구조를 갖는 블록 암호알고리즘은 F 함수의 특성에 따라 구분될 수 있다. SEED의 F 함수는 수정된 64비트 Feistel 형태로 구성된다. F 함수는 각 32비트 블록 2개(C,D)를 입력으로 받아, 32 비트 블록 2개(C',D')를 출력한다. 즉, 암호화 과정에서 64비트 블록(C,D)와 64비트 라운드 키 K_i = (k_{i0}:k_{i1})를 F함수의 입력으로 처리하여 64비트 블록(C',D')을 출력한다 (i=라운드 수)

$$
\begin{aligned}
C' &= G[G[G\{(C \oplus K_{i,0}) \oplus (D \oplus K_{i,1})\} \boxplus (C \oplus K_{i,0})] \boxplus G\{(C \oplus K_{i,0}) \oplus (D \oplus K_{i,1})\}] \\
&\quad \boxplus G[G\{(C \oplus K_{i,0}) \oplus (D \oplus K_{i,1})\} \boxplus (C \oplus K_{i,0})] \\
D' &= G[G[G\{(C \oplus K_{i,0}) \oplus (D \oplus K_{i,1})\} \boxplus (C \oplus K_{i,0})] \boxplus G\{(C \oplus K_{i,0}) \oplus (D \oplus K_{i,1})\}]
\end{aligned}
$$

< F 함수 >



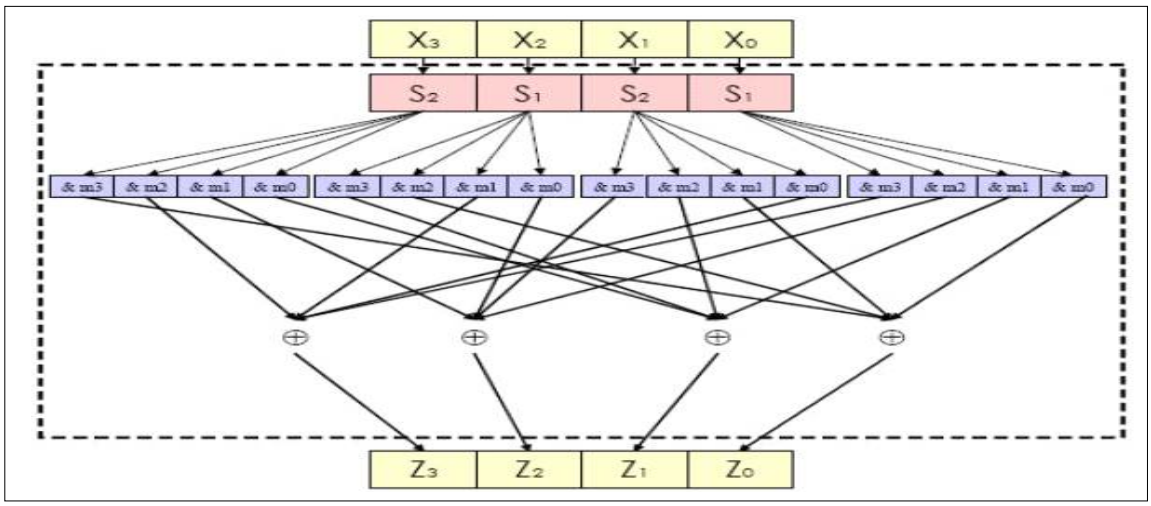
<그림 10> F함수 구조도

$$Y_3=S_2(X_3),\ Y_2=S_1(X_2),\ Y_1=S_2(X_1),\ Y_0=S_1(X_0),$$

$$Z_3=(Y_0\&m_3)\oplus(Y_1\&m_0)\oplus(Y_2\&m_1)\oplus(Y_3\&m_2)$$
$$Z_2=(Y_0\&m_2)\oplus(Y_1\&m_3)\oplus(Y_2\&m_0)\oplus(Y_3\&m_1)$$
$$Z_1=(Y_0\&m_1)\oplus(Y_1\&m_2)\oplus(Y_2\&m_3)\oplus(Y_3\&m_0)$$
$$Z_0=(Y_0\&m_0)\oplus(Y_1\&m_1)\oplus(Y_2\&m_2)\oplus(Y_3\&m_3)$$

$$(m_0=0xfc,\ m_1=0xf3,\ m_2=0xcf,\ m_3=0x3f)$$

< G 함수 >



<그림 11> G함수 구조도

3.1.5 S-BOX

G 함수의 내부에 사용되는 비선형 S-Box S_1, S_2는 다음의 식을 이용하여 생성된다.

$$S_i : Z_{2^8} \rightarrow Z_{2^8},\ S(x)=A^{(i)} \cdot x^{n_i} \oplus b_i$$

$$A^{(1)}=\begin{pmatrix}1&0&0&0&1&0&1&0\\1&1&1&1&1&1&1&0\\1&0&0&0&0&1&0&1\\0&1&0&0&0&0&1&0\\0&1&0&0&0&1&0&1\\0&0&1&0&0&0&0&1\\1&0&0&0&1&0&0&0\\0&0&0&1&0&1&0&0\end{pmatrix},\quad A^{(2)}=\begin{pmatrix}0&1&0&0&0&1&0&1\\1&0&0&0&0&1&0&1\\1&1&1&1&1&1&1&0\\0&0&1&0&0&0&0&1\\1&0&0&0&1&0&1&0\\1&0&0&0&1&0&0&0\\0&1&0&0&0&0&1&0\\0&0&0&1&0&1&0&0\end{pmatrix}$$

그림<12> S-BOX

(n_1 = 247 , n_2 = 251 , b_1 = 159 , b_2 = 56)

3.1.6 라운드 키 생성과정

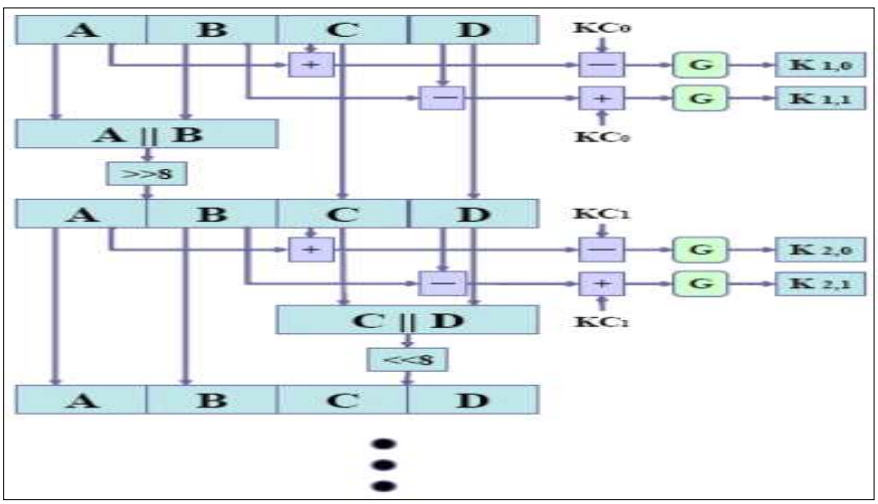
SEED의 라운드 키 생성과정은 128비트 암호키를 64비트씩 좌우로 나누어 이들을 교대로 8비트씩 좌/우로 회전이동한 후, 결과의 4워드들에 대한 간단한 산술연산과 G 함수를 적용하여 라운드 키를 생성한다. 라운드 키 생성과정은 기본적으로 하드웨어나(모든 라운드 키를 저장할 수 없는) 제한 된 자원을 갖는 스마트카드와 같은 응용에서의 효율성을 위하여, 암호화나 복호화시 암호화키로부터 필요한 라운드키를 간단히 계산할 수 있도록 설계되었다.
주어진 128비트 암호키 K = A ll B ll C ll D를 32비트 레지스터 A,B,C,D로 나눈다. 각 라운드 i에 사용되는 라운드 키 $K_i=(K_{i0}:K_{i1})$는 다음과 같은 방식으로 생성한다.

```
for( i=1; i<=16; i++) {
    K_i,0 ← G(A+C-KC_i-1);
    K_i,1 ← G(B-D+KC_i-1);
    if( i%2==1 ) A||I ← (A||B)>>>8;
    else   C||I ← (C||D)<<<8;
}
```

라 운 드 상 수	
KC_0 = 0x9e3779b9	KC_8 = 0x3779b99e
KC_1 = 0x3c6ef373	KC_9 = 0x6ef3733c
KC_2 = 0x78dde6e6	KC_{10} = 0xdde6e678
KC_3 = 0xf1bbcdcc	KC_{11} = 0xbbcdccf1
KC_4 = 0xe3779b99	KC_{12} = 0x779b99e3
KC_5 = 0xc6ef3733	KC_{13} = 0xef3733c6
KC_6 = 0x8dde6e67	KC_{14} = 0xde6e678d
KC_7 = 0x1bbcdccf	KC_{15} = 0xbcdccf1b

<그림 13> 라운드상수

<라운드 키 생성과정에 사용된 상수>



<그림 14> 라운드키 생성과정 구조

3.2 SEED 암호 소스

3.2. S-BOX

```
class Seed
{
    private $SS0 = array(
        0x2989a1a8, 0x05858184, 0x16c6d2d4, 0x13c3d3d0, 0x14445054, 0x1d0d111c
        0x1d4d515c, 0x03434340, 0x18081018, 0x1e0e121c, 0x11415150, 0x3cccf0fc
        0x28082028, 0x04444044, 0x20002020, 0x1d8d919c, 0x20c0e0e0, 0x22c2e2e0
        0x2585a1a4, 0x0f8f838c, 0x03030300, 0x3b4b7378, 0x3b8bb3b8, 0x13031310
        0x30407070, 0x0c8c808c, 0x3f0f333c, 0x2888a0a8, 0x32023230, 0x1dcdd1dc
        0x2ccce0ec, 0x15859194, 0x0b0b0308, 0x17475354, 0x1c4c505c, 0x1b4b5358
        0x24042024, 0x1c0c101c, 0x33437370, 0x18889098, 0x10001010, 0x0cccc0cc
        0x2c0c202c, 0x27c7e3e4, 0x32427270, 0x03838380, 0x1b8b9398, 0x11c1d1d0
        0x20406060, 0x10405050, 0x2383a3a0, 0x2bcbe3e8, 0x0d0d010c, 0x3686b2b4
        0x3787b3b4, 0x1a4a5258, 0x06c6c2c4, 0x38487078, 0x2686a2a4, 0x12021210
        0x21416160, 0x03c3c3c0, 0x3484b0b4, 0x01414140, 0x12425250, 0x3d4d717c
        0x1f0f131c, 0x19899198, 0x00000000, 0x19091118, 0x04040004, 0x13435350
        0x3dcdf1fc, 0x36467274, 0x2f0f232c, 0x27072324, 0x3080b0b0, 0x0b8b8388
        0x2282a2a0, 0x2e4e626c, 0x13839390, 0x0d4d414c, 0x29496168, 0x3c4c707c
        0x3f8fb3bc, 0x2fcfe3ec, 0x33c3f3f0, 0x05c5c1c4, 0x07878384, 0x14041014
        0x1eced2dc, 0x2e0e222c, 0x0b4b4348, 0x1a0a1218, 0x06060204, 0x21012120
        0x02020200, 0x35c5f1f4, 0x12829290, 0x0a8a8288, 0x0c0c000c, 0x3383b3b0
        0x3a4a7278, 0x07474344, 0x16869294, 0x25c5e1e4, 0x26062224, 0x00808080
        0x2181a1a0, 0x30003030, 0x37073334, 0x2e8ea2ac, 0x36063234, 0x15051114
        0x34c4f0f4, 0x2787a3a4, 0x05454144, 0x0c4c404c, 0x01818180, 0x29c9e1e8
        0x35053134, 0x0bcbc3c8, 0x0ecec2cc, 0x3c0c303c, 0x31417170, 0x11011110
        0x35457174, 0x3bcbf3f8, 0x1acad2d8, 0x38c8f0f8, 0x14849094, 0x19495158
        0x3fcff3fc, 0x09494148, 0x39093138, 0x27476364, 0x00c0c0c0, 0x0fcfc3cc
        0x0f0f030c, 0x0e8e828c, 0x02424240, 0x23032320, 0x11819190, 0x2c4c606c
        0x34043034, 0x31c1f1f0, 0x08484048, 0x02c2c2c0, 0x2f4f636c, 0x3d0d313c
        0x3e8eb2bc, 0x3e0e323c, 0x3c8cb0bc, 0x01c1c1c0, 0x2a8aa2a8, 0x3a8ab2b8
        0x3b0b3338, 0x1cccd0dc, 0x28486068, 0x3f4f737c, 0x1c8c909c, 0x18c8d0d8
        0x37477374, 0x2080a0a0, 0x2dcde1ec, 0x06464244, 0x3585b1b4, 0x2b0b2328
        0x23c3e3e0, 0x3989b1b8, 0x3181b1b0, 0x1f8f939c, 0x1e4e525c, 0x39c9f1f8
        0x31013130, 0x2acae2e8, 0x2d4d616c, 0x1f4f535c, 0x24c4e0e4, 0x30c0f0f0
        0x16061214, 0x3a0a3238, 0x18485058, 0x14c4d0d4, 0x22426260, 0x29092128
        0x28c8e0e8, 0x1b0b1318, 0x05050104, 0x39497178, 0x10809090, 0x2a4a6268
    );

    private $SS1 = array(
        0x38380830, 0xe828c8e0, 0x2c2d0d21, 0xa42686a2, 0xcc0fcfc3, 0xdc1eced2
        0xac2f8fa3, 0x60204060, 0x54154551, 0xc407c7c3, 0x44044440, 0x6c2f4f63
        0xc003c3c3, 0x60224262, 0x30330333, 0xb43585b1, 0x28290921, 0xa02080a0
        0xd013c3d3, 0x90118191, 0x10110111, 0x04060602, 0x1c1c0c10, 0xbc3c8cb0
        0xec2fcfe3, 0x88088880, 0x6c2c4c60, 0xa82888a0, 0x14170713, 0xc404c4c0
        0xc002c2c2, 0x44054541, 0xe021c1e1, 0xd416c6d2, 0x3c3f0f33, 0x3c3d0d31
        0x28280820, 0x4c0e4e42, 0xf436c6f2, 0x3c3e0e32, 0xa42585a1, 0xf839c9f1
        0xd818c8d0, 0x282b0b23, 0x64264662, 0x783a4a72, 0x24270723, 0x2c2f0f23
        0x40024242, 0xd414c4d0, 0x40014141, 0xc000c0c0, 0x70334373, 0x64274763
        0xf437c7f3, 0xac2d8da1, 0x80008080, 0x1c1f0f13, 0xc80acac2, 0x2c2c0c20
        0xd012c2d2, 0x080b0b03, 0xec2ecee2, 0xe829c9e1, 0x5c1d4d51, 0x94148490
        0x54174753, 0xac2e8ea2, 0x08080800, 0xc405c5c1, 0x10130313, 0xcc0dcdc1
        0xfc3fcff3, 0x7c3d4d71, 0xc001c1c1, 0x30310131, 0xf435c5f1, 0x880a8a82
        0xd011c1d1, 0x20200020, 0xd417c7d3, 0x00020202, 0x20220222, 0x04040400
        0x04070703, 0xd81bcbd3, 0x9c1d8d91, 0x98198991, 0x60214161, 0xbc3e8eb2
        0xdc1dcdd1, 0x50114151, 0x90108090, 0xdc1cccd0, 0x981a8a92, 0xa02383a3
        0x80018181, 0x0c0f0f03, 0x44074743, 0x181a0a12, 0xe023c3e3, 0xec2ccce0
        0x94168692, 0x783b4b73, 0x5c1c4c50, 0xa02282a2, 0xa02181a1, 0x60234363
        0xc808c8c0, 0x9c1e8e92, 0x9c1c8c90, 0x383a0a32, 0x0c0c0c00, 0x2c2e0e22
        0x9c1f8f93, 0x581a4a52, 0xf032c2f2, 0x90128292, 0xf033c3f3, 0x48094941
        0x14150511, 0xf83bcbf3, 0x70304070, 0x74354571, 0x7c3f4f73, 0x34350531
        0x64244460, 0x6c2d4d61, 0xc406c6c2, 0x74344470, 0xd415c5d1, 0xb43484b0
        0x74364672, 0x18190911, 0xfc3ecef2, 0x40004040, 0x10120212, 0xe020c0e0
        0xf83acaf2, 0x00010101, 0xf030c0f0, 0x282a0a22, 0x5c1e4e52, 0xa82989a1
        0x84058581, 0x14140410, 0x88098981, 0x981b8b93, 0xb03080b0, 0xe425c5e1
        0x94178793, 0xfc3cccf0, 0x1c1e0e12, 0x80028282, 0x20210121, 0x8c0c8c80
        0x74374773, 0x54144450, 0xb03282b2, 0x1c1d0d11, 0x24250521, 0x4c0f4f43
        0xec2dcde1, 0x58184850, 0x50124252, 0xe82bcbe3, 0x7c3e4e72, 0xd81acad2
        0x30300030, 0x94158591, 0x64254561, 0x3c3c0c30, 0xb43686b2, 0xe424c4e0
        0x0c0e0e02, 0x50104050, 0x38390931, 0x24260622, 0x30320232, 0x84048480
        0x34370733, 0xe427c7e3, 0x24240420, 0xa42484a0, 0xc80bcbc3, 0x50134353
        0xd819c9d1, 0x4c0c4c40, 0x80038383, 0x8c0f8f83, 0xcc0ecec2, 0x383b0b33
    );

    private $SS2 = array(
        0xa1a82989, 0x81840585, 0xd2d416c6, 0xd3d013c3, 0x50541444, 0x111c1d0d
        0x515c1d4d, 0x43400343, 0x10181808, 0x121c1e0e, 0x51501141, 0xf0fc3ccc
        0x20282808, 0x40440444, 0x20202000, 0x919c1d8d, 0xe0e020c0, 0xe2e022c2
        0xa1a42585, 0x838c0f8f, 0x03000303, 0x73783b4b, 0xb3b83b8b, 0x13101303
        0x70703040, 0x808c0c8c, 0x333c3f0f, 0xa0a82888, 0x32303202, 0xd1dc1dcd
        0xe0ec2ccc, 0x91941585, 0x03080b0b, 0x53541747, 0x505c1c4c, 0x53581b4c
        0x20242404, 0x101c1c0c, 0x73703343, 0x90981888, 0x10101000, 0xc0cc0ccc
        0x202c2c0c, 0xe3e427c7, 0x72703242, 0x83800383, 0x93981b8b, 0xd1d011c1
        0x60602040, 0x50501040, 0xa3a02383, 0xe3e82bcb, 0x010c0d0d, 0xb2b43683
        0xb3b43787, 0x52581a4a, 0xc2c406c6, 0x70783848, 0xa2a42686, 0x12101202
        0x61602141, 0xc3c003c3, 0xb0b43484, 0x41400141, 0x52501242, 0x717c3d4d
        0x131c1f0f, 0x91981989, 0x00000000, 0x11181909, 0x00040404, 0x53501343
        0xf1fc3dcd, 0x72743646, 0x232c2f0f, 0x23242707, 0xb0b03080, 0x83880b8b
        0xa2a02282, 0x626c2e4e, 0x93901383, 0x414c0d4d, 0x61682949, 0x707c3c4c
        0xb3bc3f8f, 0xe3ec2fcf, 0xf3f033c3, 0xc1c405c5, 0x83840787, 0x10141404
        0xd2dc1ece, 0x222c2e0e, 0x43480b4b, 0x12181a0a, 0x02040606, 0x21202101
        0x02000202, 0xf1f435c5, 0x92901282, 0x82880a8a, 0x000c0c0c, 0xb3b03383
        0x72783a4a, 0x43440747, 0x92941686, 0xe1e425c5, 0x22242606, 0x80800080
        0xa1a02181, 0x30303000, 0x33343707, 0xa2ac2e8e, 0x32343606, 0x11141505
        0xf0f434c4, 0xa3a42787, 0x41440545, 0x404c0c4c, 0x81800181, 0xe1e829c9
        0x31343505, 0xc3c80bcb, 0xc2cc0ece, 0x303c3c0c, 0x71703141, 0x11101101
        0x71743545, 0xf3f83bcb, 0xd2d81aca, 0xf0f838c8, 0x90941484, 0x51581949
        0xf3fc3fcf, 0x41480949, 0x31383909, 0x63642747, 0xc0c000c0, 0xc3cc0fcf
        0x030c0f0f, 0x828c0e8e, 0x42400242, 0x23202303, 0x91901181, 0x606c2c4c
        0x30343404, 0xf1f031c1, 0x40480848, 0xc2c002c2, 0x636c2f4f, 0x313c3d0d
        0xb2bc3e8e, 0x323c3e0e, 0xb0bc3c8c, 0xc1c001c1, 0xa2a82a8a, 0xb2b83a8a
        0x33383b0b, 0xd0dc1ccc, 0x60682848, 0x737c3f4f, 0x909c1c8c, 0xd0d818c8
        0x73743747, 0xa0a02080, 0xe1ec2dcd, 0x42440646, 0xb1b43585, 0x23282b0b
        0xe3e023c3, 0xb1b83989, 0xb1b03181, 0x939c1f8f, 0x525c1e4e, 0xf1f839c9
        0x31303101, 0xe2e82aca, 0x616c2d4d, 0x535c1f4f, 0xe0e424c4, 0xf0f030c3
        0x12141606, 0x32383a0a, 0x50581848, 0xd0d414c4, 0x62602242, 0x21282909
        0xe0e828c8, 0x13181b0b, 0x01040505, 0x71783949, 0x90901080, 0x62682a4a
    );

    private $SS3 = array(
        0x08303838, 0xc8e0e828, 0x0d212c2d, 0x86a2a426, 0xcfc3cc0f, 0xced2dc1e
        0x8fa3ac2f, 0x40606020, 0x45515415, 0xc7c3c407, 0x44404404, 0x4f636c2f
        0xc3c3c003, 0x42626022, 0x03333033, 0x85b1b435, 0x09212829, 0x80a0a020
        0xc3d3d013, 0x81919011, 0x01111011, 0x06020406, 0x0c101c1c, 0x8cb0bc3c
        0xcfe3ec2f, 0x88808808, 0x4c606c2c, 0x88a0a828, 0x07131417, 0xc4c0c404
        0xc2c2c002, 0x45414405, 0xc1e1e021, 0xc6d2d416, 0x0f333c3f, 0x0d313c3c
        0x08202828, 0x4e424c0e, 0xc6f2f436, 0x0e323c3e, 0x85a1a425, 0xc9f1f839
        0xc8d0d818, 0x0b23282b, 0x46626426, 0x4a72783a, 0x07232427, 0x0f232c2f
        0x42424002, 0xc4d0d414, 0x41414001, 0xc0c0c000, 0x43737033, 0x47636427
        0xc7f3f437, 0x8da1ac2d, 0x80808000, 0x0f131c1f, 0xcac2c80a, 0x0c202c2c
        0xc2d2d012, 0x0b03080b, 0xcee2ec2e, 0xc9e1e829, 0x4d515c1d, 0x84909414
        0x47535417, 0x8ea2ac2e, 0x08000808, 0xc5c1c405, 0x03131013, 0xcdc1cc0c
        0xcff3fc3f, 0x4d717c3d, 0xc1c1c001, 0x01313031, 0xc5f1f435, 0x8a82880a
        0xc1d1d011, 0x00202020, 0xc7d3d417, 0x02020002, 0x02222022, 0x04000404
        0x07030407, 0xcbd3d81b, 0x8d919c1d, 0x89919819, 0x41616021, 0x8eb2bc3e
        0xcdd1dc1d, 0x41515011, 0x80909010, 0xcccd0dc1c, 0x8a92981a, 0x83a3a023
        0x81818001, 0x0f030c0f, 0x47434407, 0x0a12181a, 0xc3e3e023, 0xcce0ec2c
        0x86929416, 0x4b73783b, 0x4c505c1c, 0x82a2a022, 0x81a1a021, 0x43636023
        0xc8c0c808, 0x8e929c1e, 0x8c909c1c, 0x0a32383a, 0x0c000c0c, 0x0e222c2e
        0x8f939c1f, 0x4a52581a, 0xc2f2f032, 0x82929012, 0xc3f3f033, 0x49414809
        0x05111415, 0xcbf3f83b, 0x40707030, 0x45717435, 0x4f737c3f, 0x05313435
        0x44606424, 0x4d616c2d, 0xc6c2c406, 0x44707434, 0xc5d1d415, 0x84b0b434
        0x46727436, 0x09111819, 0xcef2fc3e, 0x40404000, 0x02121012, 0xc0e0e020
        0xcaf2f83a, 0x01010001, 0xc0f0f030, 0x0a22282a, 0x4e525c1e, 0x89a1a829
        0x85818405, 0x04101414, 0x89818809, 0x8b93981b, 0x80b0b030, 0xc5e1e425
        0x87939417, 0xcccf0fc3c, 0x0e121c1e, 0x82828002, 0x01212021, 0x8c808c0c
        0x47737437, 0x44505414, 0x82b2b032, 0x0d111c1d, 0x05212425, 0x4f434c0f
        0xcde1ec2d, 0x48505818, 0x42525012, 0xcbe3e82b, 0x4e727c3e, 0xcad2d81a
        0x00303030, 0x85919415, 0x45616425, 0x0c303c3c, 0x86b2b436, 0xc4e0e424
        0x0e020c0e, 0x40505010, 0x09313839, 0x06222426, 0x02323032, 0x84808404
        0x07333437, 0xc7e3e427, 0x04202424, 0x84a0a424, 0xcbc3c80b, 0x43535013
        0xc9d1d819, 0x4c404c0c, 0x83838003, 0x8f838c0f, 0xcec2cc0e, 0x0b33383b
    );
);
```

3.2.2 round function

```
    private function ConvertInt($float) {
        $UnsignedIntMax = PHP_INT_MAX;
        $UnsignedIntMin = ( PHP_INT_MAX * -1 ) -1;
        if(is_float($float) &&  $float < $UnsignedIntMin ) {
            $division = floor($float / $UnsignedIntMin );
            $n = ($division % 2 == 0)?0:$UnsignedIntMin;
            if( $float < $UnsignedIntMin ) $c = $float - ( $UnsignedIntMin * $
        }
        elseif(is_float($float) && $float > $UnsignedIntMax) {
            $division = floor($float / $UnsignedIntMax );
            $n = ($division % 2 == 0)?0:$UnsignedIntMax;
            if( $float > $UnsignedIntMax) $c = $float - ( $UnsignedIntMax * $d
        }
        else $c = $float;
        return $c;
    }

    public function SeedRound(
        &$L0, &$L1,             // [in, out] left-side variable at each round
        &$R0, &$R1,             // [in] right-side variable at each round
        $K = array()            // [in] round keys at each round
    )
    {
        $T0 = $R0 ^ $K[0];
        $T1 = $R1 ^ $K[1];
        $T1 ^= $T0;

        $T1 = $this->SS0[$this->GetB0($T1)] ^ $this->SS1[$this->GetB1($T1)] ^
        $T0 += $T1;
        $T0 = $this->ConvertInt($T0);

        $T0 = $this->SS0[$this->GetB0($T0)] ^ $this->SS1[$this->GetB1($T0)] ^
        $T1 += $T0;
        $T1 = $this->ConvertInt($T1);

        $T1 = $this->SS0[$this->GetB0($T1)] ^ $this->SS1[$this->GetB1($T1)] ^
        $T0 += $T1;
        $T0 = $this->ConvertInt($T0);

        $L0 ^= $T0;
        $L1 ^= $T1;
    }
```

3.2.3 encrtyption function

```
    public function SeedEncrypt(
        $pbData = array(),                  // [in] data to be encrypted
        $pdwRoundKey = array(),             // [in] round keys for encryption
        &$outData = array()                 // [out] encrypted data
    )
    {
        $L0 = 0x0;
        $L1 = 0x0;
        $R0 = 0x0;
        $R1 = 0x0;
        $K = array();
        $nCount = 0;

        // Set up input values for encryption
        $L0 = ( $pbData[0] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[1] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[2] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[3] & 0x000000ff );

        $L1 = ( $pbData[4] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^ ( $pbData[5] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^ ( $pbData[6] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^ ( $pbData[7] & 0x000000ff );

        $R0 = ( $pbData[8] & 0x000000ff );
        $R0 = ( $R0 <<8 ) ^ ( $pbData[9] & 0x000000ff );
        $R0 = ( $R0 <<8 ) ^ ( $pbData[10] & 0x000000ff );
        $R0 = ( $R0 <<8 ) ^ ( $pbData[11] & 0x000000ff );

        $R1 = ( $pbData[12] & 0x000000ff );
        $R1 = ( $R1 <<8 ) ^ ( $pbData[13] & 0x000000ff );
        $R1 = ( $R1 <<8 ) ^ ( $pbData[14] & 0x000000ff );
        $R1 = ( $R1 <<8 ) ^ ( $pbData[15] & 0x000000ff );

        // Reorder for little endian
        // Because java virtual machine use big endian order in default
        if (!$this->ENDIAN) {
            $this->EndianChange($L0);
            $this->EndianChange($L1);
            $this->EndianChange($R0);
            $this->EndianChange($R1);
        }

        $K[0] = $pdwRoundKey[$nCount++];
        $K[1] = $pdwRoundKey[$nCount++];
        $this->SeedRound($L0, $L1, $R0, $R1, $K); /*   1 */
```

```
$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    2 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    3 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    4 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    5 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    6 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    7 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    8 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    9 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*   10 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*   11 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*   12 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*   13 */


$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*  14 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*  15 */

$K[0] = $pdwRoundKey[$nCount++];
$K[1] = $pdwRoundKey[$nCount++];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*  16 */

if (!$this->ENDIAN)  {
    $this->EndianChange($L0);
    $this->EndianChange($L1);
    $this->EndianChange($R0);
    $this->EndianChange($R1);
}

// Copying output values from last round to outData
for ($i=0; $i<16; $i++) $outData[$i] = null;
for ($i=0; $i<4; $i++)
{
    $outData[$i] = ( ( ( $R0 ) >>( 8 * ( 3 - $i ) ) ) & 0xff );
    $outData[4+$i] = ( ( ( $R1 ) >> ( 8 * ( 3 - $i ) ) ) & 0xff );
    $outData[8+$i] = ( ( ( $L0 ) >> ( 8 * ( 3 - $i ) ) ) & 0xff );
    $outData[12+$i] = ( ( ( $L1 ) >> ( 8 * ( 3 - $i ) ) ) &0xff );
}
```

3.2.4 decrtyption function

```
// Same as encrypt, except that round keys are applied in reverse order
    public function SeedDecrypt(
            $pbData = array(),                          // [in] encrypted data
            $pdwRoundKey = array(),                     // [in] round keys for decrypt
            &$outData = array()                         // [out] data to be encrypt
    )
    {
        $L0 = 0x0;
        $L1 = 0x0;
        $R0 = 0x0;
        $R1 = 0x0;
        $K = array();
        $nCount = 31;

        // Set up input values for decryption
        $L0 = ( $pbData[0] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[1] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[2] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[3] & 0x000000ff );

        $L1 = ( $pbData[4] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^(  $pbData[5] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^ ( $pbData[6] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^ ( $pbData[7] & 0x000000ff );
```

```
// Same as encrypt, except that round keys are applied in reverse order
    public function SeedDecrypt(
            $pbData = array(),                          // [in] encrypted data
            $pdwRoundKey = array(),                 // [in] round keys for decrypt
            &$outData = array()                         // [out] data to be encrypt
    )
    {
        $L0 = 0x0;
        $L1 = 0x0;
        $R0 = 0x0;
        $R1 = 0x0;
        $K = array();
        $nCount = 31;

        // Set up input values for decryption
        $L0 = ( $pbData[0] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[1] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[2] & 0x000000ff );
        $L0 = ( $L0 << 8 ) ^ ( $pbData[3] & 0x000000ff );

        $L1 = ( $pbData[4] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^(  $pbData[5] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^ ( $pbData[6] & 0x000000ff );
        $L1 = ( $L1 << 8 ) ^ ( $pbData[7] & 0x000000ff );

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    2 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    3 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    4 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    5 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    6 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    7 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*    8 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*    9 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*   10 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*   11 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*   12 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*   13 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*   14 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount--];
$this->SeedRound($L0, $L1, $R0, $R1, $K); /*  15 */

$K[1] = $pdwRoundKey[$nCount--];
$K[0] = $pdwRoundKey[$nCount];
$this->SeedRound($R0, $R1, $L0, $L1, $K); /*  16 */

if (!$this->ENDIAN) {
    $this->EndianChange($L0);
    $this->EndianChange($L1);
    $this->EndianChange($R0);
    $this->EndianChange($R1);
}
```

```
 // Copy output values from last round to outData
     for ($i=0; $i<16; $i++) $outData[$i] = null;
     for ($i=0; $i < 4; $i++)
     {
         $outData[$i] = ( ( ( $R0 ) >> ( 8 * ( 3 - $i ) ) ) & 0xff );
         $outData[4+$i] = ( ( ( $R1 ) >> ( 8 * ( 3 - $i ) ) ) & 0xff );
         $outData[8+$i] = ( ( ( $L0 ) >> ( 8 * ( 3 - $i ) ) ) & 0xff );
         $outData[12+$i] = ( ( ( $L1 ) >> ( 8 * ( 3 - $i ) ) ) & 0xff );
       }
 }
```

3.2.5 라운드키

```
public function SeedRoundKey(
        &$pdwRoundKey = array(),            // [out] round keys for encryp
        $pbUserKey = array()                // [in]    secret user key
)
{
    $K = array();
    $nCount = 2;

    // Set up input values for Key Schedule
    $A = ( $pbUserKey[0] & 0x000000ff );
    $A = ( $A << 8 ) ^ ( $pbUserKey[1] & 0x000000ff );
    $A = ( $A << 8 ) ^ ( $pbUserKey[2] & 0x000000ff );
    $A = ( $A << 8 ) ^ ( $pbUserKey[3] & 0x000000ff );

    $B = ( $pbUserKey[4] & 0x000000ff );
    $B = ( $B<<8 ) ^ ( $pbUserKey[5] & 0x000000ff );
    $B = ( $B<<8 ) ^ ( $pbUserKey[6] & 0x000000ff );
    $B = ( $B<<8 ) ^ ( $pbUserKey[7] & 0x000000ff );

    $C = ( $pbUserKey[8] & 0x000000ff );
    $C = ( $C << 8 ) ^ ( $pbUserKey[9] & 0x000000ff );
    $C = ( $C << 8 ) ^ ( $pbUserKey[10] & 0x000000ff );
    $C = ( $C << 8 ) ^ ( $pbUserKey[11] & 0x000000ff );

    $D = ( $pbUserKey[12] & 0x000000ff );
    $D = ( $D << 8 ) ^ ($pbUserKey[13] & 0x000000ff );
    $D = ( $D << 8 ) ^ ( $pbUserKey[14] & 0x000000ff );
    $D = ( $D << 8 ) ^ ( $pbUserKey[15] & 0x000000ff );

    // reorder for little endian
    if (!$this->ENDIAN)  {
        $A = $this->EndianChange($A);
        $B = $this->EndianChange($B);
        $C = $this->EndianChange($C);
        $D = $this->EndianChange($D);
    }

    $T0 = $A + $C - $this->KC[0];
    $T0 = $this->ConvertInt($T0);

    $T1 = $B - $D + $this->KC[0];
    $T1 = $this->ConvertInt($T1);

    $pdwRoundKey[0] = $this->SS0[$this->GetB0($T0)] ^ $this->SS1[$this->Ge
    $pdwRoundKey[1] = $this->SS0[$this->GetB0($T1)] ^ $this->SS1[$this->Ge

$this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 1 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate1($K, $A, $B, $C, $D, 2 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 3 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate1($K, $A, $B, $C, $D, 4 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];
```

```
$this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 5 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate1($K, $A, $B, $C, $D, 6 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 7 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate1($K, $A, $B, $C, $D, 8 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 9 );
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate1($K, $A, $B, $C, $D, 10);
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 11);
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];

$this->EncRoundKeyUpdate1($K, $A, $B, $C, $D, 12);
$pdwRoundKey[$nCount++] = $K[0];
$pdwRoundKey[$nCount++] = $K[1];


        $this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 13);
        $pdwRoundKey[$nCount++] = $K[0];
        $pdwRoundKey[$nCount++] = $K[1];

        $this->EncRoundKeyUpdate1($K, $A, $B, $C, $D, 14);
        $pdwRoundKey[$nCount++] = $K[0];
        $pdwRoundKey[$nCount++] = $K[1];

        $this->EncRoundKeyUpdate0($K, $A, $B, $C, $D, 15);
        $pdwRoundKey[$nCount++] = $K[0];
        $pdwRoundKey[$nCount++] = $K[1];
    }
    public function SeedRoundKeyText( &$pdwRoundKey, $pbUserKey ){
        $Data = array();
        for($i=0;$i<strlen($pbUserKey);$i++ ) $Data[$i] = ord($pbUserKey{$i});
        $this->SeedRoundKey( $pdwRoundKey, $Data );
    }
    public function SeedEncryptText( $pbData, $pdwRoundKey, &$outData )
    {
        $Data = array();
        for($i=0;$i<strlen($pbData);$i++ ) $Data[$i] = ord($pbData{$i});
        $this->SeedEncrypt( $Data, $pdwRoundKey, $outData );
    }
    public function SeedDecryptText( $pbData, $pdwRoundKey, &$outData )
    {
        $Data = array();
        for($i=0;$i<strlen($pbData);$i++ ) $Data[$i] = ord($pbData{$i});
        $this->SeedDecrypt( $Data, $pdwRoundKey, $outData );
    }
}
```

3.2.6 게시판 암호화 소스

```
<?
  include 'class.crypto.php';
  $crypto = new Crypto();

if($userid && $userid!=$item_id)
        {
?>
                        <?= $crypto->encrypt($item_content) ?>
<?
        }
?>
```

암호화전 평문

<그림 15> 암호화 전 게시판

<그림 15> 는 게시판에 암호화 소스를 적용하기 전 사진이다.

암호문

<그림 16> 암호화 후 게시판

<그림 16>은 게시판에 암호화 소스를 적용시킨 후 사진이다.

4. Snort

4.1 침입탐지시스템

4.1.1 침입탐지시스템이란

네트워크을 통해 네트워크 시스템 자원을 오용하거나 비정상적인 행위로 데이터를 훼손하거나 서비스 불능상태로 만드는 보안 위협들을 탐지하여 그에 대응할 수 있는 시스템을 말한다. 현재의 외부 인터넷으로부터 보안 위협에 대해 Firewall 시스템으로 보안을 구축하고 있으나 내부자에 의한 침입에 대해서는 방어할 수 없는 상태이며 보안 사고에 대한 즉각적인 보고, 이벤트의 연동, 법적 증거분석을 위한 보안 감사, 그리고 자동화된 대응체계가 미비한 상태이다. 침입 탐지 시스템은 네트워크 트래픽 또는 서버에 대해 의심스러운 행위를 감시 추적하며 감지된 행동에 따라 경고를 발생하거나 위협적인 행위를 차단할 수 있다.

4.1.2 침입탐지 시스템의 종류

침입탐지 시스템은 침입 탐지를 하는 데이터 소스에 따라 호스트 기반 침입탐지시스템과 multi 호스트 기반 침입탐지시스템 그리고 네트워크 기반 침입탐지시스템으로 구분된다. 침입탐지의 모델에 따라서 오용(Misuse) 탐지시스템과 변칙(anomaly) 탐지시스템으로 구분된다.

①오용 탐지

오용탐지 모델은 다른 말로 Signature-based 탐지 모델이라고도 한다. 이는 알려진 시스템의 취약점을 이용하는 침입방법의 패턴이나 특정 수단에 대한 signature를 정의하고 그에 따른 트래픽을 비교함으로써 침입을 탐지하는 방식이다.

②변칙 탐지

변칙 탐지 모델은 다르게 Profile-based 탐지 모델이라고도 한다. 이는 네트워크 상의 사용자의 행동을 수집하고 분석하여 통계적 분석에 의해 변칙적 행위를 감지하게 된다.

	Misuse	Anomaly
장점	시스템 자원의 비중이 적다. 탐지 확률이 높다.	보편적인 통계적 처리 방법을 이용가능 Misuse방식과 비교하여 보안 인적 자원 비중이 적다.
단점	신규 해킹 출현 시 마다 새로운 signature 반영이 필요Signature 관리를 위한 보안 전문인력이 필요함.	시스템 자원의 비중이 요구됨 통계적 기준을 정함으로 탐지 결과의 확실성이 떨어진다.

①호스트 기반 침입탐지시스템

이 시스템은 단일 호스트에 설치되어 해당 시스템의 감사 데이타를 사용해 침입을 탐지하는 시스템이다. 이방식은 대부분 해당 시스템의 파일들에 대한 무결성을 점검한다. 즉, 그 호스트의 중요한 파일이나 보안 관련 파일들이 수정되었는지 또는 임의의 사용자가 사용자 자신의 보안 수준을 넘는 파일들로 접근하려 시도했는지를 감지한다. 호스트 기반의 침입탐지 시스템은 해당 호스트에 설치되므로 시스템 별로 호환성의 문제를 갖고 있으며 시스템 자원에 부하를 줄 수 있는 단점이 있다. 대신 네트워크 환경에 구애 받지 않는 장점을 가지고 있다.

②네트워크 기반 침입탐지시스템

네트워크 기반의 침입탐지 시스템들은 통상 전 네트워크 세그먼트를 감시하는 전용의 시스템들이다. 대부분의 경우 Firewall 외부 네트워크 세그먼트나 내부의 주요 네트워크 세그먼트에 설치한다. 네트워크 기반의 IDS는 네트워크 상에 흘러다니는 모든 패킷들을 검사하여 알려진 공격들이나 의심스러운 행동에 대하여 분석하게 된다. 호스트 기반 IDS와 달리 네트워크 세그먼트를 감시하므로 효과적인 침입탐지 환경을 갖출 수 있고 보호하는 네트워크 서버들의 호환성과 상관없이 기존 서비스의 중단 및 영향없이 시스템을 구축할 수 있게 한다. 다만 네트워크 상의 흘러다니는 모든 트래픽을 검사하므로 네트워크 트래픽 용량에 영향을 많이 받는 단점이 있다.

	호스트 기반	네트워크 기반
장점	콘솔 작업자의 공격을 차단	저렴한 비용으로 효과적인 보안 처리가 가능 대규모 네트워크 지원 호스트 공격 전에 탐지 가능
단점	관리와 유지보수가 어려움 대규모 네트워크 지원이 곤란 호스트 성능에 영향을 미침	콘솔 작업자의 공격은 탐지 못함

4.2.3 침입탐지시스템의 기능

①경보기능

해커에 의한 네트워크나 시스템의 탐지 그리고 직접적인 공격에 대하여 경보, 이 메일 발송, 또는 SNMP trap발송, 기타 다른 방법들을 이용하여 관리자에게 통보한다. 이러한 사전 통보를 통하여 네트워크 관리자 및 시스템 관리자는 능동적으로 전산 자원의 손실을 사전에 방어할 수 있게 된다.

②세션 차단기능

침입탐지시스템은 의심스러운 행위를 감지하면 해당 세션을 차단하는 기능을 제공한다. 이는 다른 서비스에 영향을 주지 않고 효과적으로 해당 공격 시도를 선별적으로 막아주는 기능이다. 이 기능은 공격 대상 호스트로 접근 중인 네트워크 세션을 감지한 후 침입탐지 시스템이 그 공격대상 시스템을 가장하여 공격자에게 네트워크 세션을 종료하게끔 한다. 일반적으로 Telnet, FTP와 같은 TCP 세션에 대해서는 TCP FIN 또는 RST(reset) 패킷을 공격자에게 전달하여 세션이 끊어지게 하며 SNMP 또는 Name service와 같은 UDP 세션(UDP에서 세션은 모순되는 말이지만)은 ICMP의 "port unreachable" 메시지를 전달하여 더 이상의 진행을 차단하게 된다.

③셔닝(shunning) 기능

이 기능은 침입탐지시스템 자체로 의심스러운 탐지나 공격을 차단하는 것이 아니라 라우터나 Firewall에게 공격자나 의심스러운 자의 근원지 IP 주소와 서비스 포트에 대해 해당 패킷을 막도록 지시하는 기능이다. 그렇게 함으로써 그 다음부터 오는 동일 패킷들은 폐기된다. 이러한 연동 기능은 일반적으로 발생할 수 있는 프로토콜의 오류에 대해서도 민감하게 동작하기 때문에 그 효용성에 대해 조심스럽게 검토하여야 하며 충분하게 네트워크 트래픽에 대한 검토 후 적용되어야 한다.

또한 라우터에 ACL(Access Control List)를 적용하게 되는 경우 라우터의 성능을 저하시켜 높은 트래픽을 가지고 있는 환경에서는 서비스에 지장을 초래할 수도 있다.

④사용자 프로그램의 실행
일부 침입탐지시스템의 경우 지정된 이벤트에 따라 특정한 사용자 프로그램을 실행할 수 있도록 한다. 공격자로부터 탐지나 공격에 대하여 적절한 대응 방안에 대해 사용자 환경에 맞게 적용할 수 있으므로 유용한 기능이며 필요하다면 해킹 발생에 대하여 시스템의 순서적인 정지를 가능하게 할 수 있다.

4.2.4 침입탐지시스템의 장단점

장점 :침입탐지시스템은 기존 네트워크의 보안 레벨을 한 층 높여줄 수 있는 시스템이며 외부 또는 내부의 보안 위협에 대해 수동적인 대응에서 보다 적극적이고 능동적인 대응 방안을 마련하도록 한다.
일반적인 네트워크 공격은 오랜 시간의 탐지행동을 거친 후 보안 홀을 찾아 공격으로 전이하게 된다. 따라서 침입탐지시스템의 도입은 이러한 공격 전 단계의 행위를 감지함으로써 실질적인 네트워크 공격을 무산시킬 수 있는 강력한 무기를 제공하게 된다.

단점 :일반적으로 침입탐지시스템들의 민감성이 매우 중요하다. 특정 탐지나 공격의 패턴을 민감하게 감지할 수 있어야 고도의 네트워크 해킹 기법들에 대응할 수 있다. 그렇지만 이러한 민감성은 일반적으로 발생할 수 있는 트래픽들, 예를 들어 일부 NAT(Network Address Translation) 장비에서 발생하는 port 0 와 같은 이벤트나 한 기업에서 프록시 캐시를 사용하는 경우는 "port scan" 이라는 특정의 이벤트들로 감지할 수도 있다. 이를 False Positive라고 한다. 말 그대로 하면 "잘못된 양성반응" 이라고 할 수 있다. 이러한 False positive는 모든 침입탐지시스템에서 일반적인 현상이므로 충분한 customize가 필요하게 된다. 따라서 세션 차단 기능 또는 라우터나 Firewall 연동 시에 주의 깊게 정의를 하여야 하며 만약 이런 False positive에 대해 정의하지 않는 경우 일반적인 서비스에 큰 장애를 유발하게 된다.

4.2 Snort 설치

4.2.1 Snort 설치

시작하기전 yum install gcc*을 이용해 gcc를 설치
https://snort.org/ 에 들어가 snort설치에 필요한 snort/libnet/libdnet/daq를 다운

```
snort-2.9.7.0.tar.gz
libnet-1.0.2a.tar.gz
libdnet-1.12.tgz
daq-2.0.4.tar.gz
```

tar xvzf를 이용해 다운 받은 4개의 압축을 풀어준다.

```
[root@localhost install]# tar xvzf libnet-1.0.2a.tar.gz
[root@localhost install]# tar xvzf libdnet-1.12.tgz
[root@localhost install]# tar xvzf snort-2.9.7.0.tar.gz
[root@localhost install]# tar xvzf daq-2.0.4.tar.gz
```

압축을 푼후 libnet 으로 이동

```
[root@localhost install]# cd ./Libnet-1.0.2a/
[root@localhost Libnet-1.0.2a]# ./configure
[root@localhost Libnet-1.0.2a]# make && make install
```

cd.. 으로 이동하여

```
[root@localhost install]# cd ./libdnet-1.12
[root@localhost libdnet-1.12]# ./configure
[root@localhost libdnet-1.12]# make && make install
```

위와 똑같이 한후 cd..

```
[root@localhost install]# cd ./daq-2.0.4
[root@localhost daq-2.0.4]# ./configure
```

cd.. 으로 이동후

```
[root@localhost install]# cd ./snort-2.9.7.0
[root@localhost snort-2.9.7.0]# ./configure
```

Snort 설치가 끝난후 Snort -V로 설치확인

```
[root@localhost snort-2.9.7.0]# snort -V

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.7.0 GRE (Build 149)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.4.0
           Using PCRE version: 7.8 2008-09-05
           Using ZLIB version: 1.2.3

You have mail in /var/spool/mail/root
```

4.2.2 오류

1.pcre 오류

```
ERROR!  Libpcre header not found.
Get it from http://www.pcre.org
```

위와 같은 pcre 오류가 있을시 yum install을 통해 pcre-devel 설치

```
[root@localhost snort-2.9.7.0]# yum install -y pcre-devel
```

2.libpcap 오류

```
ERROR! Libpcap library/headers (libpcap.a (or .so)/pcap.h)
```

->다음과 같은 에러 발생시 libpcap 설치해줘야 하다.

```
[root@localhost install]# yum install -y libpcap-devel
```

3.flex 와 bison

```
configure: error: Your operating system's lex is insufficient to compile
         libsfbpf.  You should install both bison and flex.
         flex is a lex replacement that has many advantages,
         including being able to compile libsfbpf.  For more
         information, see http://www.gnu.org/software/flex/flex.html .
```

다음과 같은 에러는 yum을 이용해 flex와 bison을 설치

4.zlib 에러

```
ERROR!  zlib header not found, go get it from
http://www.zlib.net
```

위 에러는 yum을 이용해 zlib-devel을 설치

5.libpcap 오류

daq에서 ./configure 또는 make&&maek install 시 아래와 같은 오류가 나올때가 있다.
libpcap는 www.tcpdump.org 에서 최신버젼으로 설치가 가능하다.
libpcap최신 버전을 설치후 ./configure 와 make&&make install을 해준다.
그러나 여기까지 했어도 오류가 뜰 것이다. 그 경우 daq에서 libpcap를 찾을 때
/usr/lib에서 찾는데 실제로는 저장이 /usr/local/lib에 되어있기 때문이다.

```
error libpcap library version = 1.0.0 not found
```

→libpcap 오류

```
[root@localhost lib]# ls
libpcap.a  libpcap.so  libpcap.so.1  libpcap.so.1.7.3
[root@localhost lib]# cp ./*libpcap /usr/lib
```

cd /usr/local/lib 으로 이동해

```
[root@localhost lib]# cp ./libpcap.* /usr/lib
```

libpcap으로 시작하는 모든 파일을 usr/lib 으로 옮겨줍니다.

4.2.3 Snort-Rule 설치

```
[root@localhost install]# mkdir /etc/snort
[root@localhost install]# mkdir /var/log/snort
[root@localhost install]# cd /etc/snort
```

mkdir /etc/snort -> 환경설정을 할 폴더
mkdir /var/log/snort -> log 파일이 저장될 폴더

```
[root@localhost snort]# tar xvzf /install/snortrules-snapshot-2956.tar.gz
-C /etc/snort
```

tar xvzf /다운로드 폴더/snortrules-snapshot-2907.tar.gz -C /etc/snort -> 룰셋파일을 스노트에 설정

```
[root@localhost snort]# ls
etc  preproc_rules  rules  so_rules
[root@localhost snort]# ls -l ./etc/
합계 4480
-rw-r--r--. 1 1210 1210    3854 2014-12-12 01:38 classification.config
-rw-r--r--. 1 1210 1210     746 2014-12-12 01:38 reference.config
-rw-r--r--. 1 1210 1210 4483836 2014-12-12 01:41 sid-msg.map
-rw-r--r--. 1 1210 1210   29395 2014-12-12 01:38 snort.conf
-rw-r--r--. 1 1210 1210    2556 2014-12-12 01:38 threshold.conf
-rw-r--r--. 1 1210 1210   53841 2014-12-12 01:38 unicode.map
[root@localhost snort]# cp etc/* /etc/snort
```

/etc/snort/etc 안의 설정파일을 /etc/snort 로 복사

[root@localhost snort]# groupadd snort -> snort 그룹 생성

[root@localhostsnort]#useradd-gsnortsnort ->snort 그룹에 snort 계정 생성

[root@localhostsnort]#chownsnort:snort/var/log/snort ->폴더의 소유권자와 소유그룹을 snort 로 변경

[root@localhostsnort]#touch/var/log/snort/alert -> alert 빈 파일 생성

[root@localhostsnort]#chownsnort:snort/var/log/snort/alert -> 파일의 소유권자와 소유그룹을 snort 로 변경

[root@localhostsnort]#chmod600/var/log/snort/alert -> 파일의 권한변경

[root@localhost snort]# mkdir /usr/local/lib/snort_dynamicrules

[root@localhostsnort]#cp/etc/snort/so_rules/precompiled/Centos-5-4/i386/2.9.5.6/*.so/usr/local/lib/snort_dynamicrules
-> snort 에 사용되는 라이브러리를 복사

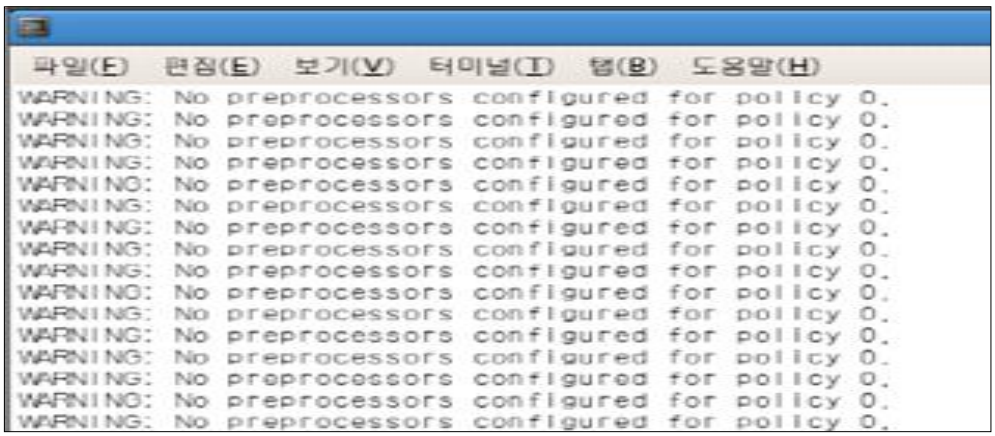
[root@localhost snort]# cat /etc/snort/so_rules/*.rules >> /etc/snort/rules/so-rules.rules
->so_rules의 rules파일을 스노트가 사용할 경로 /etc/snort/rules/so-rules.rules 파일에 덮어쓴다

```
root@localhost snort]# vi /etc/snort/snort.conf
```

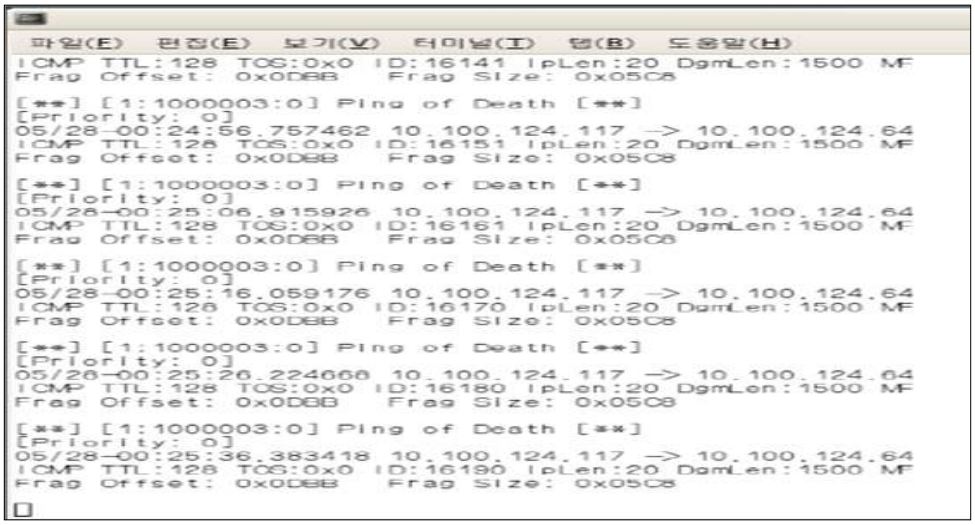
-> 환경설정 파일

```
104 var RULE_PATH ../rules
105 var SO_RULE_PATH ../so_rules
106 var PREPROC_RULE_PATH ../preproc_rules
107
108 # If you are using reputation preprocessor set these
109 var WHITE_LIST_PATH ../rules
110 var BLACK_LIST_PATH ../rules
```

경로가 맞는지 확인



<그림 17> Snort-Rule 작동 확인



<그림 18> 룰 설정으로 필터링 된 로그 기록

5. 결론

우리가 일반적으로 인터넷이라고 알고 있는 웹은 과거에는 PC가 있는 장소에서만 사용할 수 있었지만, 스마트폰이 개발된 이후에는 장소에 구애받지 않고 스마트폰을 통해서도 사용할 수 있게 됐다. 여기까지는 많은 사람들이 알고 있는 내용이지만 인터넷 익스플로러, 크롬, 사파리와 같은 인터넷 브라우저 뿐만이 아니라 카카오톡, 라인, 다음 카페, 모바일 게임 등과 같은 모바일앱들 또한 모두 웹을 통해 통신이 이루어지고 있다는 사실을 알고 있는 사람은 그리 많지 않다. 스마트폰에서 사용되는 모바일앱들은 각각 모양은 다르지만, 모두 기존의 인터넷과 동일한 웹을 기반으로 통신하고 있다. 결국 우리가 사용 중인 스마트폰이나 태블릿의 통신은 모두 웹을 통해 이루어지고 있고 이런 모바일 기기들이 대중화된 지금, 웹은 더이상 우리와 뗄레야 뗄 수 없는 관계가 되었다. 웹의 대중화와 더불어 기존의 오프라인 서비스들도 하나둘씩 온라인 서비스를 지원하게 됐다. 그 결과 현재는 인터넷 뱅킹과 같은 금융 거래부터 등록

증 발급 등의 민원 처리까지 다수의 업무들이 모두 웹을 통해 서비스가 가능해졌다. 웹은 기존의 편리한 서비스라는 개념에서 벗어나 이제 우리 생활의 일부가 되었다.
하지만 웹이 대중화 되고 많은 기업들이 웹을 통해 서비스를 제공하면서 기업의 자산을 노리는 사이버 공격 또한 웹을 주요 타겟으로 삼게 됐다. 이는 최근 이슈가 되고 있는 해킹들이 대부분 웹을 통해 이루어지고 있는 사실을 통해서도 확인할 수 있다. 웹은 기업의 중요 자산으로 연결되는 통로와 같기 때문에 웹이 공격 당하는 순간 개인정보 유출, 금전적 피해, 내부 시스템 파괴등 심각한 2차 피해로 이어질 수 있다. 또한 웹 공격의 경우 검색엔진에서 조금만 검색해봐도 손쉽게 공격을 시도할 수 있는 툴과 공격 방법까지 친절히 가르쳐 주고 있는 동영상들을 확인할 수 있어, 웹 공격에 대해 잘 모르는 사람도 매우 쉽게 따라할 수 있기 때문에 그 위험도는 매우 치명적이다. 많은 기업들이 네트워크 방화벽이나 IDS/IPS와 같은 네트워크 보안 장비들을 중요하게 생각하고 이를 구비함에 따라 공격자들이 시스템에 직접 접근하기는 어려워졌다. 하지만 기업이 서비스를 제공하는 통로인 웹 서비스는 열어놓을 수밖에 없고, 이는 3년간 웹 애플리케이션에 대한 침해공격이 가장 많다는 결과를 통해 웹 보안의 중요성을 깨달을 수 있다. 하지만 많은 경영자들, 보안관리자들이 늘 새로운 보안 트렌드에 대해서 궁금해하면서도 정작 이 웹 보안을 잘 구축한 경우는 많지 않다. 웹이 생활의 일부가 된 상황에서 웹 보안은 선택이 아닌 필수이다.
접근성이 용이한 웹 서버의 보안 기능을 보강하기 위해 웹서버와 SNORT를 연동시켜 위협신호를 탐지하는 한편 위협에 대비하여 주요 파일의 자료를 암호화하여 자료가 유출되더라도 내용 확인을 거부함으로써 자료를 안전하게 보호할 수 있는 시스템을 구현 보안 웹서버 구현과정에서 IDS의 기능과 필요성, 자료 암호화 중요성 등을 재인식 할 수 있는 시간이 되었습니다.

6. 참고자료

(1)PHP 프로그래밍 입문 - 한빛아카데미

(2)뇌를 자극하는 redhat fedora - 한빛미디어

(3)리눅스 쉘 스크립트 프로그래밍 입문 - 제이펍

(4)인터넷해킹과 보안 - 한빛아카데미

(5)유닉스 이론과 실습 - 한빛미디어

7. PPT

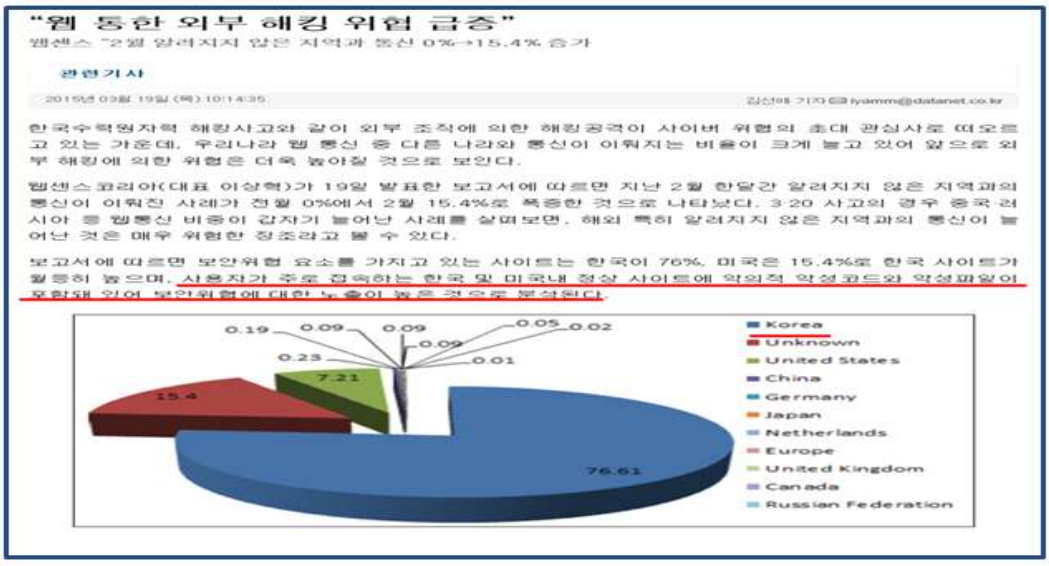

SEED암호를 이용한
보안 웹 서버
2015. 6. 9
지도교수: 양환석교수님
5조 GhostService24


목 차
• 조원 소개 및 역할
• 주제 선정 이유
• 추진 경과
• 시스템 구상도
• 개발 및 운영
• 결론

조원 소개 및 역할

조장	10 박철오	총괄 및 감독
조원	10 전인혁	웹 스크립트 개발
	10 양유빈	IDS 구축
	11 김은지	Web server DB 구축 SEED 암호 적용
	12 김지현	Web server 구축

주제 선정 이유(1/2)



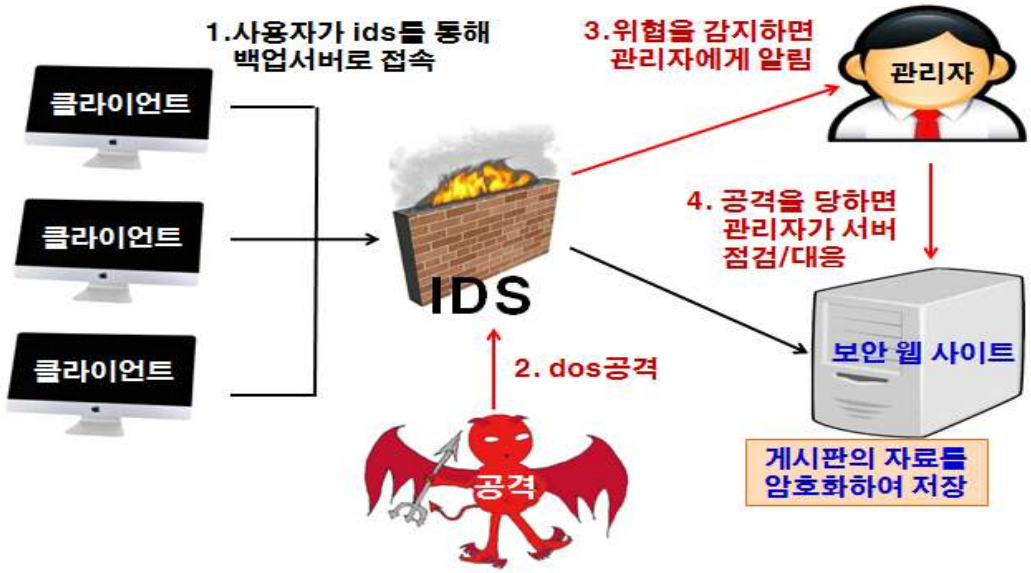
주제 선정 이유(2/2)

- 웹 서버는 접근성이 용이한 반면 보안에 취약하기 때문에 외부 해킹 위협이 급증
- 이에 대한 보안대책으로
  - IDS를 활용하여 위협신호를 자동 탐지하고
  - 탐지에 대비, 주요 자료는 SEED 암호를 이용하여 암호화하는 등의 방법으로 보안대책을 구현

추진 경과

구 분	2014년			2015년				
	10월	11월	12월	1월	2월	3월	4월	5월
주제 선정								
자료 수집								
시스템 구현 연구								
Web server 구축								
IDS 구축								
Seed 암호 적용								
웹 스크립트 제작								
시스템 점검/보완								

시스템 구상도(1/2)



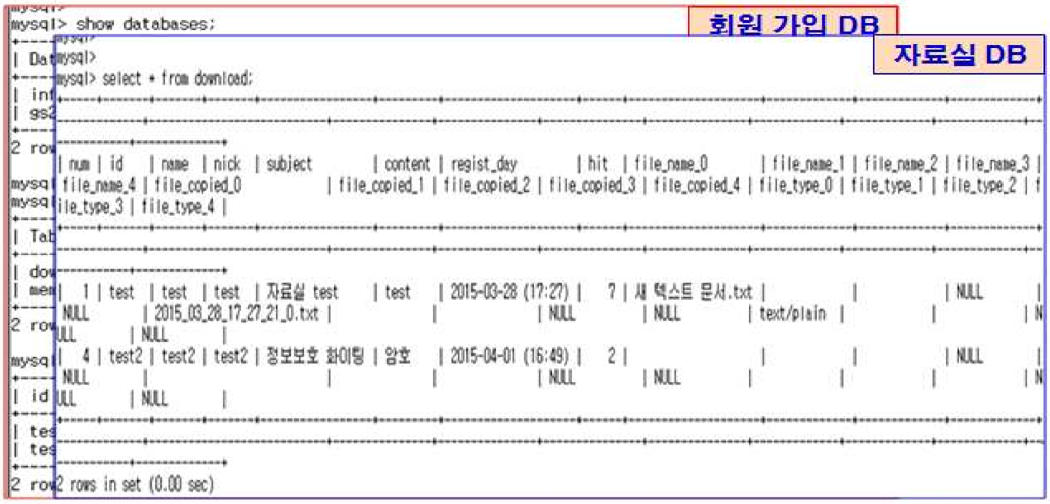
시스템 구상도(2/2)

개발 및 운영환경

- ❍ 개발도구 ⇨ php
- ❍ 사용 소스 ⇨ Snort , seed암호
- ❍ 운영체제 ⇨ Windows 2003 , CentOS 5.0

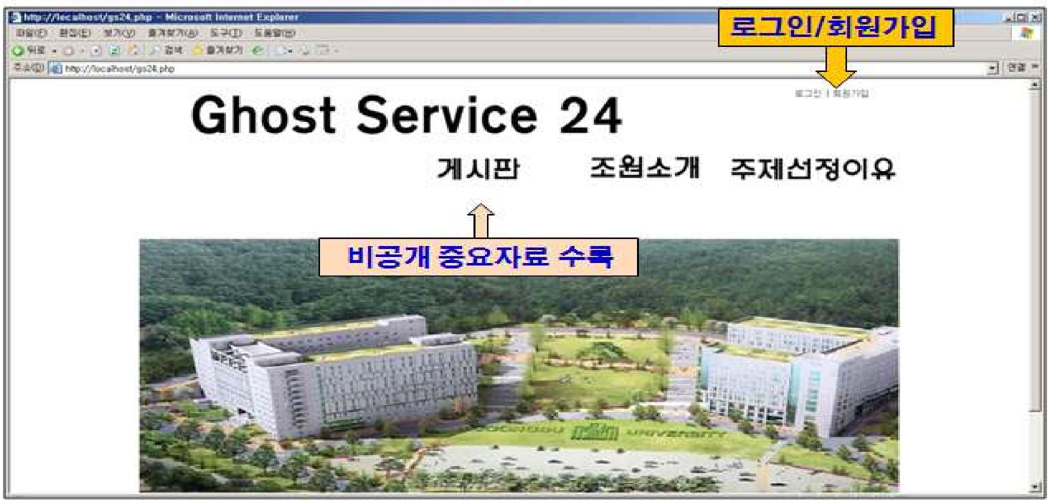
개발 및 운영(1/10)

웹서버 구축



개발 및 운영(2/10)

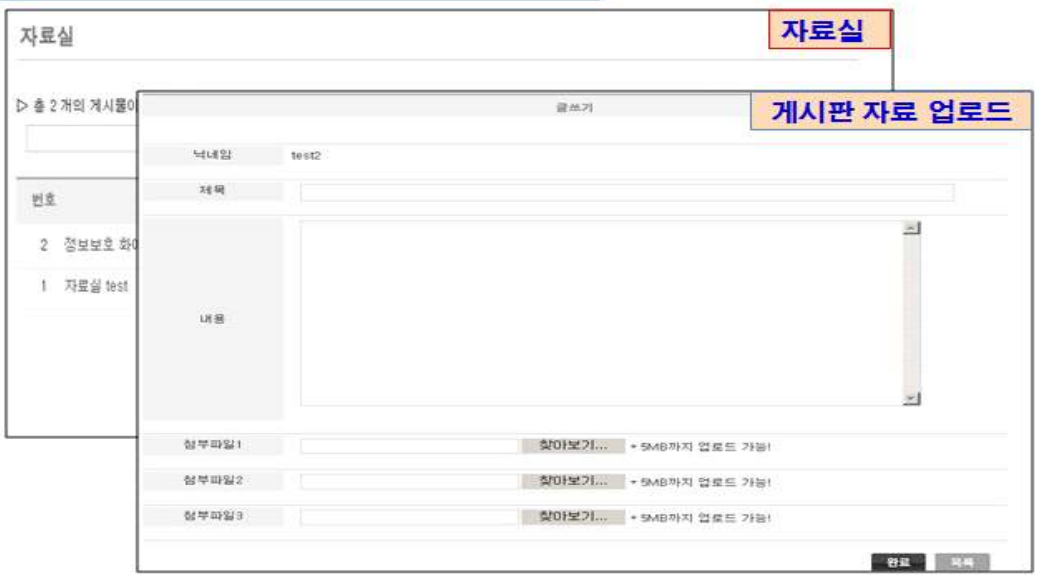
웹서버 홈페이지 제작



개발 및 운영(3/10)

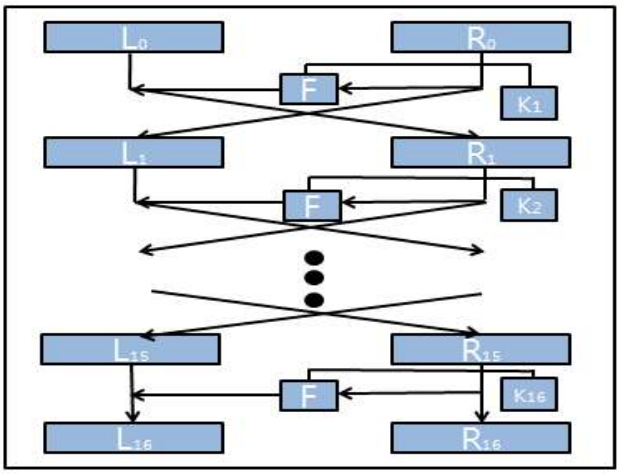
웹서버 운영절차

개발 및 운영(4/10)



개발 및 운영(5/10)

SEED 암호



- Feistel 구조
- 128비트 암호문 생성
- F함수는 K_i 에서 키값을 받아 64비트 블록 출력
- G함수를 이용한 라운드 키(K_i)생성
- G함수는 S-BOX를 이용

개발 및 운영(6/10)

SEED 암호 적용

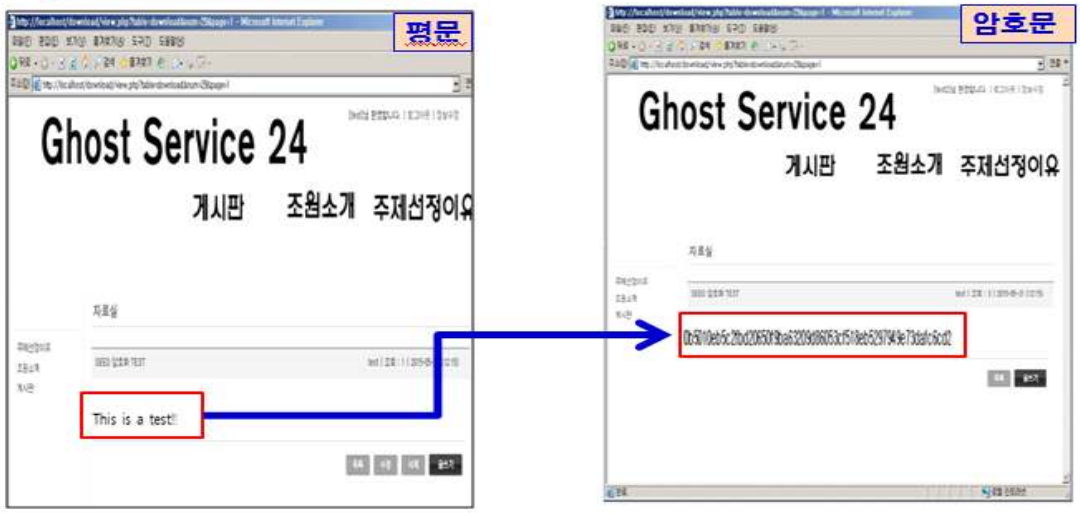
암호화 소스

로그인

```
/************************** SEED encrtyption function ************
    public function SeedEncrypt(
        $pbData = array(),                          // [in] data to be
/************************** SEED decrtyption function ************
// Same as encrypt, except that round keys are applied in reverse
    public function SeedDecrypt(
            $pbData = array(),                          // [in] encrypted data
            $pdwRoundKey = array(),                     // [in] round keys for de
            &$outData = array()                         // [out] data to be en
    )
    {
        $t0 = 0x0;
        $t1 = 0x0;
        $R0 = 0x0;
        $R1 = 0x0;
        $K = array();
        $ncount = 31;

        // Set up input values for decryption
        $t0 = ( $pbData[0] & 0x000000ff );
        $t0 = ( $t0 << 8 ) ^ ( $pbData[1] & 0x000000ff );
        $t0 = ( $t0 << 8 ) ^ ( $pbData[2] & 0x000000ff );
        $t0 = ( $t0 << 8 ) ^ ( $pbData[3] & 0x000000ff );

        $t1 = ( $pbData[4] & 0x000000ff );
        $t1 = ( $t1 << 8 ) ^( $pbData[5] & 0x000000ff );
        $t1 = ( $t1 << 8 ) ^ ( $pbData[6] & 0x000000ff );
        $t1 = ( $t1 << 8 ) ^ ( $pbData[7] & 0x000000ff );

        $R0 = ( $pbData[8] & 0x000000ff );
        $R0 = ( $R0 << 8 ) ^ ( $pbData[9] & 0x000000ff );
        $R0 = ( $R0 << 8 ) ^ ( $pbData[10] & 0x000000ff );
        $R0 = ( $R0 << 8 ) ^ ( $pbData[11] & 0x000000ff );

        $R1 = ( $pbData[12] & 0x000000ff );
        $R1 = ( $R1 << 8 ) ^ ( $pbData[13] & 0x000000ff );
        $R1 = ( $R1 << 8 ) ^ ( $pbData[14] & 0x000000ff );
        $R1 = ( $R1 << 8 ) ^ ( $pbData[15] & 0x000000ff );

        // Reorder for little endian
        if (!$this->ENDIAN) {
            $this->EndianChange($t0);
            $this->EndianChange($t1);
            $this->EndianChange($R0);
            $this->EndianChange($R1);
        }
```

개발 및 운영(7/10)

SEED 암호 적용



개발 및 운영(8/10)

SNORT

- ♦ SNORT는 전형적인 네트워크 침입탐지시스템
- ♦ 네트워크에서 보안을 위협하는 위해 행위가 발생할 경우 이를 탐지, 경보
- ♦ 기존 탐지된 위해 행위에 대하여서 후속 공격시 탐지 대응이 가능하나 새로운 공격에 대한 방어 기능은 제한

개발 및 운영(9/10)

SNORT 연동

개발 및 운영(10/10)

결 론

- 접근성이 용이한 웹 서버의 보안 기능을 보강하기 위해
  - 웹서버와 SNORT를 연동시켜 위협신호를 탐지하는 한편
  - 위협에 대비하여 주요 파일의 자료를 암호화하여 자료가 유출되더라도 내용 확인을 거부함으로써 자료를 안전하게 보호할 수 있는 시스템을 구현
- 보안 웹서버 구현과정에서 IDS의 기능과 필요성, 자료 암호화 중요성 등을 재인식

Q & A

Thank you