sql injection = 웹 파라미터 입력값 검증을 하지 않아 데이터베이스의 정보를 가져오는것
XXS = 사용자의 브라우저 정보를 가져와서 쿠키재사용공격 & 악성코드 배포목적

웹 해킹 이해 - 파일 업로드 취약점

파일업로드 취약점 (공격) -
<조건>웹쉘파일이 정상 업로드 -> 서버애서 사이트스크립트애서 화이트리스트방식으로 차단/ 허용되어야할 확장자만!(jpg, png, pdf...)
/올린웹쉘의 절대경로를 알아야한다/ ->download.php?id=4&board_id=1 파일이 다운로드/데이터베이스에서 경로와 파일명 을 가져옴
올라간 웹쉘스크립트가 실행이 가능해야함->php.ini 에서 스크립트실행권한업로드 디렉토리에 제거

파일 업로드 취약점 공격 이해

파일 업로드 취약점 공격 이란?

- 웹 서비스 첨부 파일, 환경 설정 미흡을 이용하여 악의적인 스크립트가 포함된 파일을 업로드 한 후에 웹 서버에 침투를 하는 공격
- 공격자는 서버 사이드 스크립트(PHP, JSP, .NET 등)을 이용하여 웹쉘(WebShell)을 제작
- 웹쉘(WebShell)은 원격에서 웹 서버를 제어하기 위한 목적으로 만들어졌으나, 지금은 웹쉘 = 악성코드라고 분류를 하여 안티 바이러스에서 탐지함
- 게시판 첨부 파일, 이력서 첨부 파일, 이미지 첨부 파일, 웹 채팅방 파일 공유 기능 등에서 발생
- 직접 만들어 사용할 수 있으나, https://github.com/tennc/webshell 같은 깃허브에서 쉽게 구할 수 있음

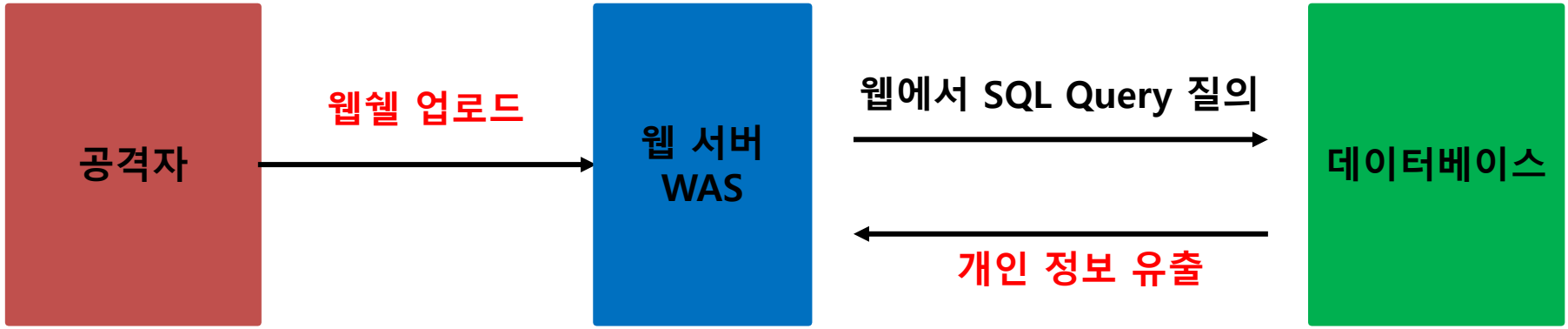
파일업로드 취약점 (공격)
RC취약점 (원격 명령어 실행 공격)

파일 업로드 취약점 공격의 목적

웹 서버를 통해 데이터베이스의 정보를 획득

- 데이터베이스에 직접 공격을 할 수 없기 때문에, 웹 서버를 침투한 후 소스코드내 데이터베이스 연결 정보를 통해 개인 정보 쿼리(Query)
- 웹 서버를 통해 데이터베이스의 2차 공격 진행도 가능함

3티어구조

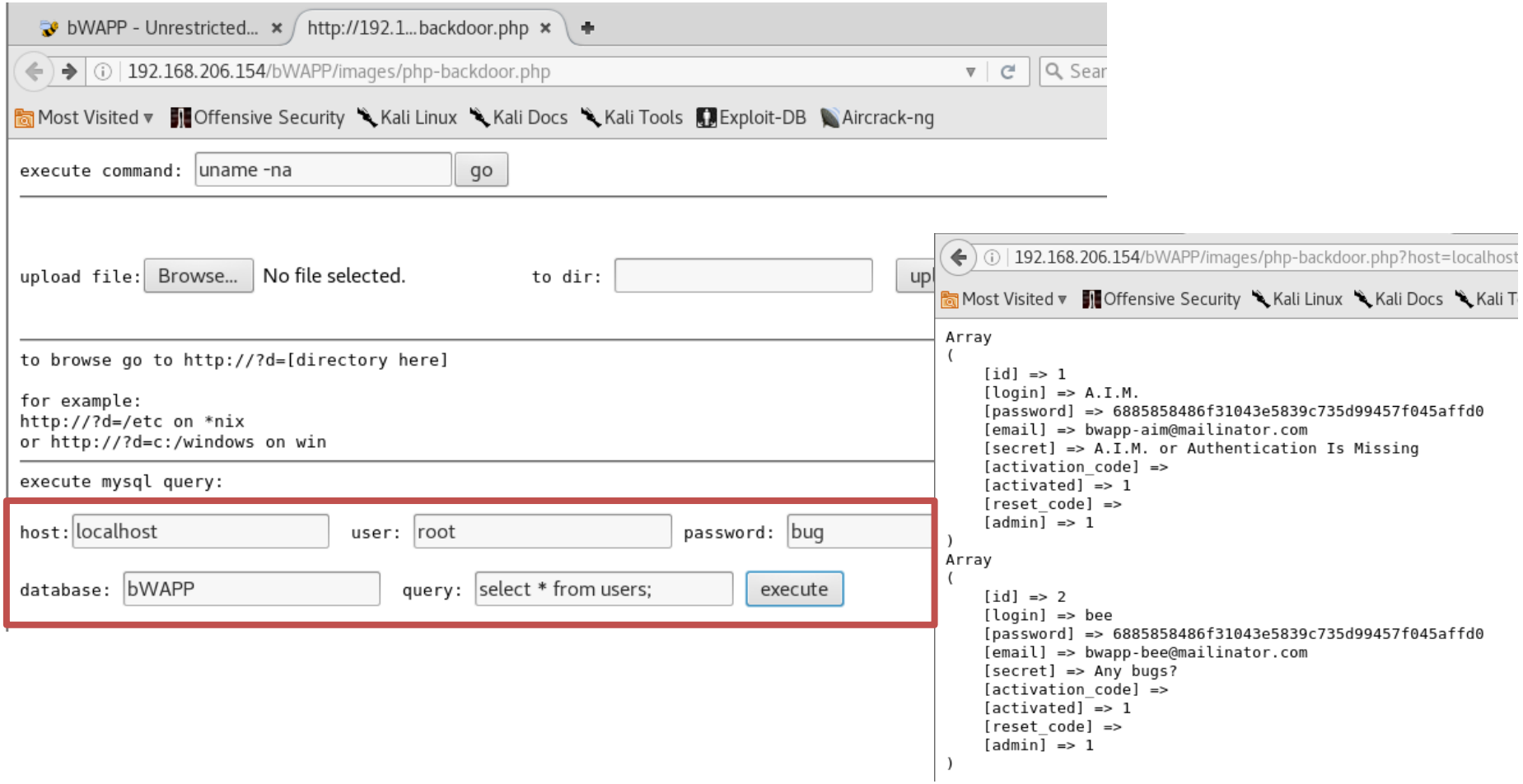


php 환경 -> php, php3, php5, phtml(대소문자도 시도!)
asp환경 -> asp, asa, asx, cer(인증서 확장자)
jsp 환경 -> jsp, js%70

파일 업로드 취약점 공격 이해

웹 서버를 통해 데이터베이스의 정보를 획득

- 웹 서버를 통해 데이터베이스의 주요 정보를 쿼리하여 가져옴



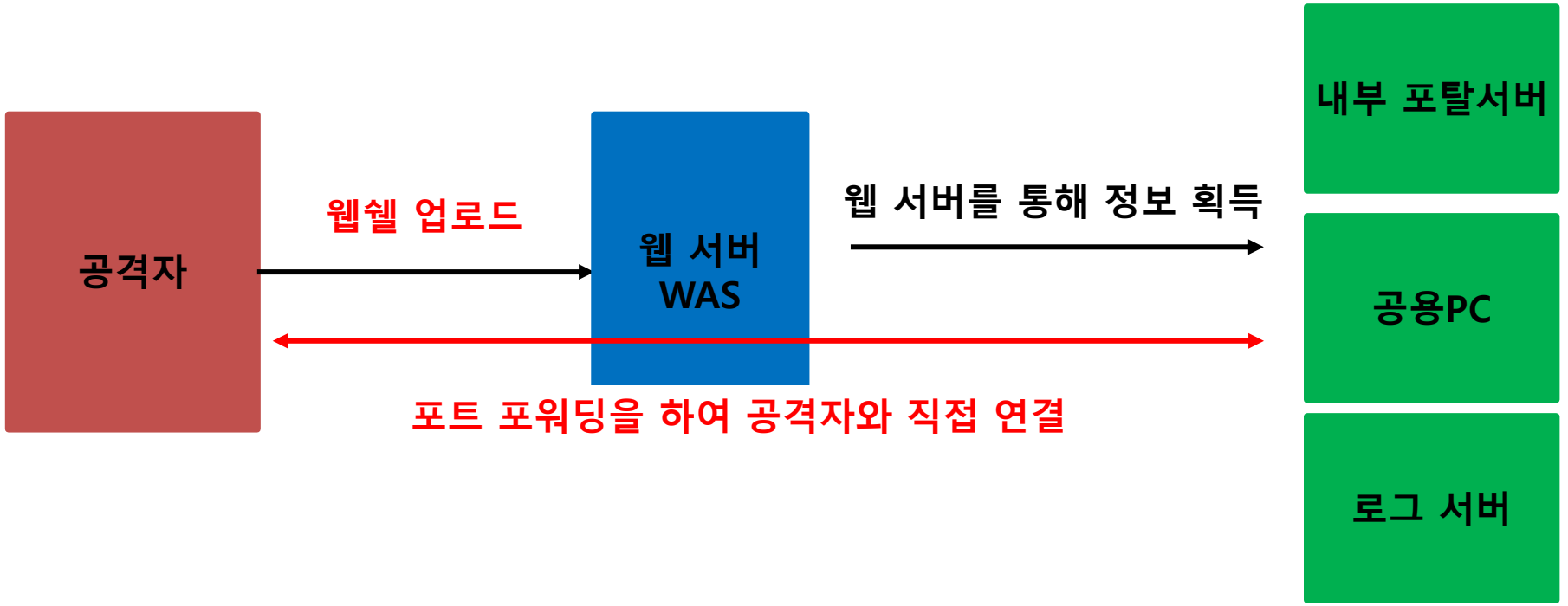
파일 업로드 취약점 공격의 목적

웹 서버를 시작으로 근접 네트워크 침투

netstat -na

ps -a

- 데이터베이스에 직접 공격을 할 수 없기 때문에, 웹 서버를 침투한 후 내부 시스템의 정보 획득
- 내부 포탈 서버, 로그 서버 등 내부 시스템을 대상으로 포트포워딩(Port Forwarding), 터널링 기법을 통해 공격자와 직접 연결을 함

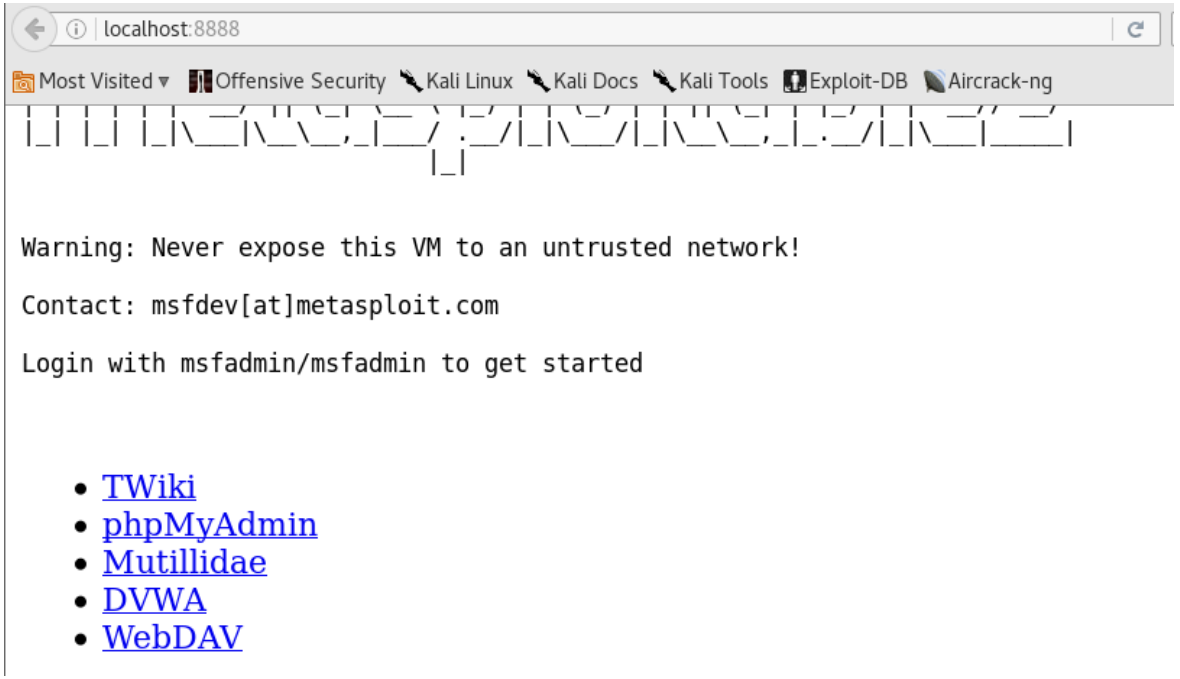


파일 업로드 취약점 공격의 목적

웹 서버를 시작으로 근접 네트워크 침투

- 내부 포탈 서버, 로그 서버 등 내부 시스템을 대상으로 포트포워딩(Port Forwarding), 터널링 기법을 통해 공격자와 직접 연결을 함

```
meterpreter > portfwd add -l 8888 -p 80 -r 192.168.206.133
[*] Local TCP relay created: :8888 <-> 192.168.206.133:80
```

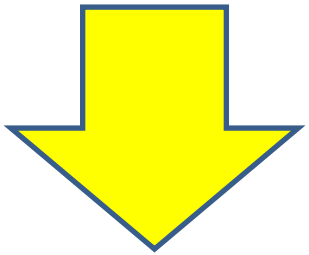


파일 업로드 취약점 대응 방안

시큐어 코딩 - 소스코드 레벨 대응

- 첨부 파일 확장자 검증은 서버 사이드 스크립트에서 검증해야 함
- 첨부파일 확장자 검증은 "블랙 리스트의 차단"이 아닌 "화이트 리스트의 허용"으로 해야 함

```
function file_upload_check_1($file, $file_extensions  = array("asp", "aspx", "dll", "exe", "jsp", "php"), $directory = "images")
```



블랙 리스트-> 화이트리스트

```
function file_upload_check_2($file, $file_extensions  = array("jpeg", "jpg", "png", "gif"), $directory = "images")
```

파일 업로드 취약점 대응 방안

시큐어 코딩 - 소스코드 레벨 대응 (PHP)

```php
<?php
        $tmp_name = $_FILES['Filedata']['tmp_name'];
        $filename = $_FILES['Filedata']['name'];
        $filename_ext = strtolower(array_pop(explode('.',$filename)));
        $allow_ext = array("jpg", "png", "hwp", "pptx", "docx", "xlsx", "pdf");
        if(!in_array($filename_ext, $allow_ext)) {
                echo "허용되지않는 확장자 파일입니다.";
                exit;
        }
        // 파일 이름의 예: "../upload/20191109231417212120083.hwp"
        $newPath = '../upload/'.date('YmdHis').mt_rand().'.'.$filename_ext;
        @move_uploaded_file($tmp_name, $newPath);
?>
```

파일 업로드 취약점 대응 방안

시큐어 코딩 - 소스코드 레벨 대응 (JSP)

```
… 생략 …
<%
String savePath="/var/www/uploads"; //업로드 디렉터리
int sizeLimit = 5 * 1024 * 1024; //업로드 파일 사이즈 제한

try
MultipartRequest multi=new MultipartRequest(request, savePath,
sizeLimit, "euc-kr", new DefaultFileRenamePolicy());
  Enumeration formNames=multi.getFileNames(); //폼의 이름 반환
  String formName=(String)formNames.nextElement();
  String filename=multi.getFilesystemName(formName); //파일의 이름 얻기
  String file_ext = filename.substring(filename.lastIndexOf('.') + 1);

//모든 문자를 소문자로 치환
ext = ext.toLowCase();
//허용할 파일 확장자를 기준으로 유효성 검사
if(!( file_ext.equalsIgnoreCase("doc") || file_ext.equalsIgnoreCase("hwp") ||  file_ext.equalsIgnoreCase("pdf")) )
    out.print("업로드 금지 파일");

  if(filename == null)
    out.print("파일 업로드 실패");
  else
    filename=new String(filename.getBytes("8859_1"),"euc-kr"); //한글 인코딩

  out.print("File Name :" + filename);
catch(Exceptione)
… 생략 …
```

파일 업로드 취약점 대응 방안

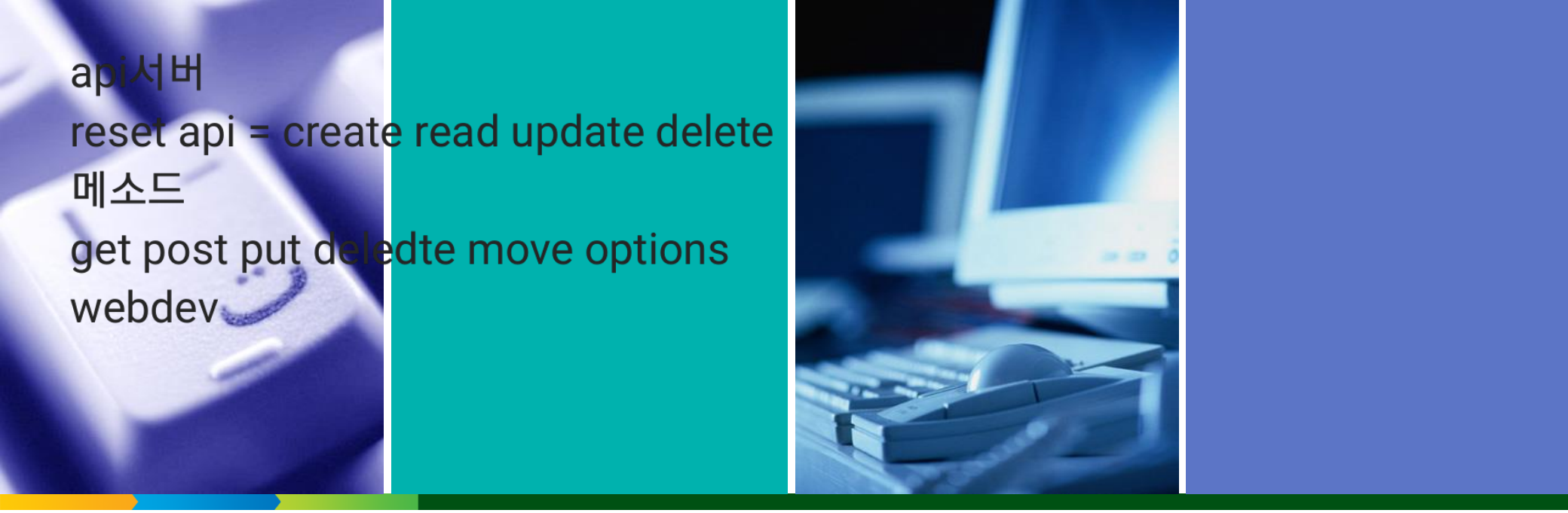
파일 업로드 취약점 대응 방안 - 시스템 보안 레벨

- Apache 설정 파일인 httpd.conf에 해당 디렉토리에 대한 문서 타입을 컨트롤하기 위해 Directory 섹션의 AllowOverride 지시자에서 FileInfo 또는 All 추가

```
#httpd.conf 파일 내용
AccessFileName .htaccess
<Directory "/var/www/htdocs/upload/">
AllowOverride FileInfo
</Directory>
```

- 파일 업로드 디렉토리에 .htaccess 파일을 만들고 다음과 같이 AddType 지시자를 이용, 현재 서버에서 운영되는 Server Side Script 확장자를 text/html로 MIME Type을 재조정하여 업로드 된 Server Side Script가 실행되지 않도록 설정

```
#.htaccess파일 내용
<.htaccess>
<FilesMatch “\.(ph|inc|lib)”>
Order allow, deny
Deny from all
</FilesMatch>
AddType text/html .html .htm .php .php3 .php4 .phtml .phps .in .cgi .pl .shtml .jsp
```



api서버
reset api = create read update delete
메소드
get post put deledte move options
webdev

Thank You !