비 공 개

국가보훈부
소스코드 보안약점 진단 보고서

2024. 11.

< 주의 사항 >

◇ 본 보고서는 일부 오탐율이 존재할 수 있으므로 참고자료로 활용하시기 바랍니다.

◇ 조치 시 한국인터넷진흥원에 있는 기술안내서 가이드에 정보보호 시스템 안전의 가이드를 참고하시기 바랍니다.

※ https://www.kisa.or.kr/2060204/form?postSeq=5

◇ 조치여부 확인 요청 시 조치된 소스코드를 암호화 압축(.zip)하여 nTOPS 로 접수해주시기 바랍니다.

행정안전부

국가정보자원관리원
디지털안전상황실

❑ 점검목적

❍ 목적 : 서비스 예정 및 운영 중인 홈페이지의 소스코드 보안약점을 진단 후 보안취약점 제거하여 사이버보안 위협 안정성 확보

❑ 점검개요

❍ 대상 : 이어온

❍ 점검기간 : 2024 년 11 월 29 일

❍ 조치지원 헬프데스크 연락처 : (042) 863-5016

❑ 점검결과

❍ 점검 결과요약(Top 10)

진단번호	보안약점	탐지건수	비고
CWE-754, ERR06-J	부적절한 예외 처리	42	위험도 등급 : 4
CWE-674	종료되지 않는 반복문 또는 재귀 함수	32	위험도 등급 : 4
CWE-306, CWE-287	적절한 인증없는 중요 기능 허용	31	위험도 등급 : 5
CWE-489	제거되지 않고 남은 디버그 코드	12	위험도 등급 : 4
CWE-209, CWE-497	오류메시지를 통한 정보노출	7	위험도 등급 : 3
CWE-488	잘못된 세션에 의한 데이터 정보 노출	5	위험도 등급 : 4
CWE-404, CWE-772	부적절한 자원 해제	5	위험도 등급 : 4
CWE-390, ERR00-J	오류 상황 대응 부재	4	위험도 등급 : 4
CWE-330	적절하지 않은 난수 값 사용	4	위험도 등급 : 5
CWE-476, EXP01-J	널(Null) 포인터 역참조	4	위험도 등급 : 5
합계		146	

조치결과 회신

진단번호	조치내용 및 결과회신	비고
1	ex) ① 제거완료	-
2	② 실제 소스코드 확인결과 문제없음	-
3	③ 조치불가(조치불가 사유 입력)	-

1 소스코드 보안약점 진단 결과

1. 프로젝트 정보

항목	내용
프로젝트명	국가보훈부 이어온
소스코드 파일 수	598개
소스코드 라인 수	199207 라인
총 보안약점 개수	154 개 (전체 검출 449개 중 조치 0개, 무시 295개)
프로젝트 위험도	5 (높을수록 위험 1..5)
리비전 정보	2024년 11월 29일 AM 11시 27분
보고서 작성일자	2024년 11월 29일 PM 4시 05분

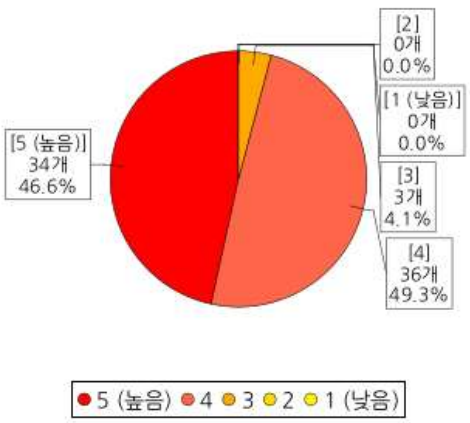
2. 분석 결과 요약

2.1. 파일별 보안약점 분포

위험도별 보안약점 검출 파일 개수					안전한 파일 개수	합계
5 (높음)	4	3	2	1 (낮음)		
34	36	3	0	0	525	598
5.7%	6.0%	0.5%	0.0%	0.0%	87.8%	100%

전체파일 보안약점 비율

보안약점 파일 위험도 비율

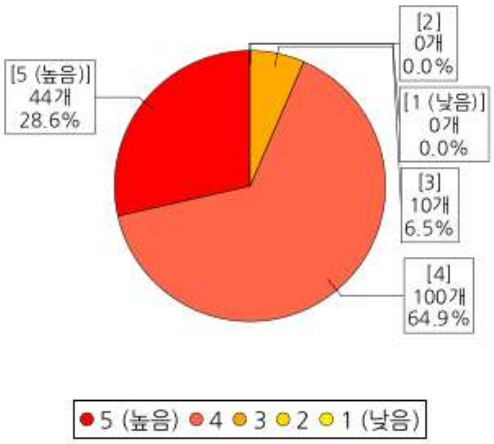


* 무시 처리된 보안약점은 제외된 결과입니다.

2.2. 보안약점별 위험도 분포

5 (높음)	4	3	2	1 (낮음)	합계
44	100	10	0	0	154
28.6%	64.9%	6.5%	0.0%	0.0%	100%

보안약점 위험도 비율



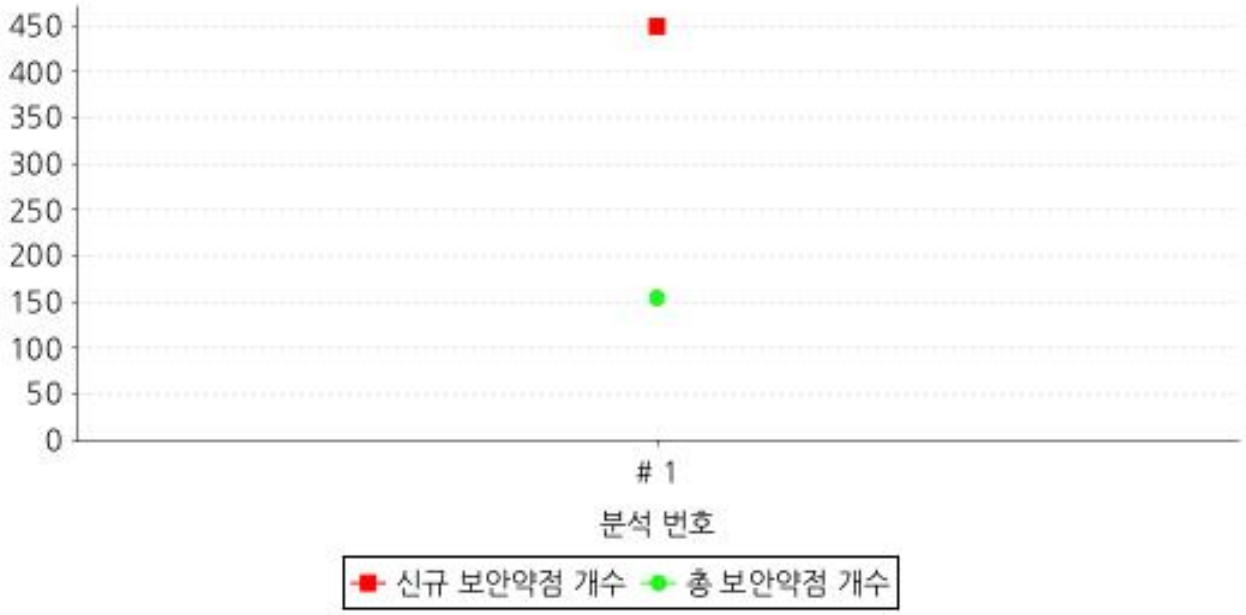
* 무시 처리된 보안약점은 제외된 결과입니다.

2.3. 보안약점 Top 5

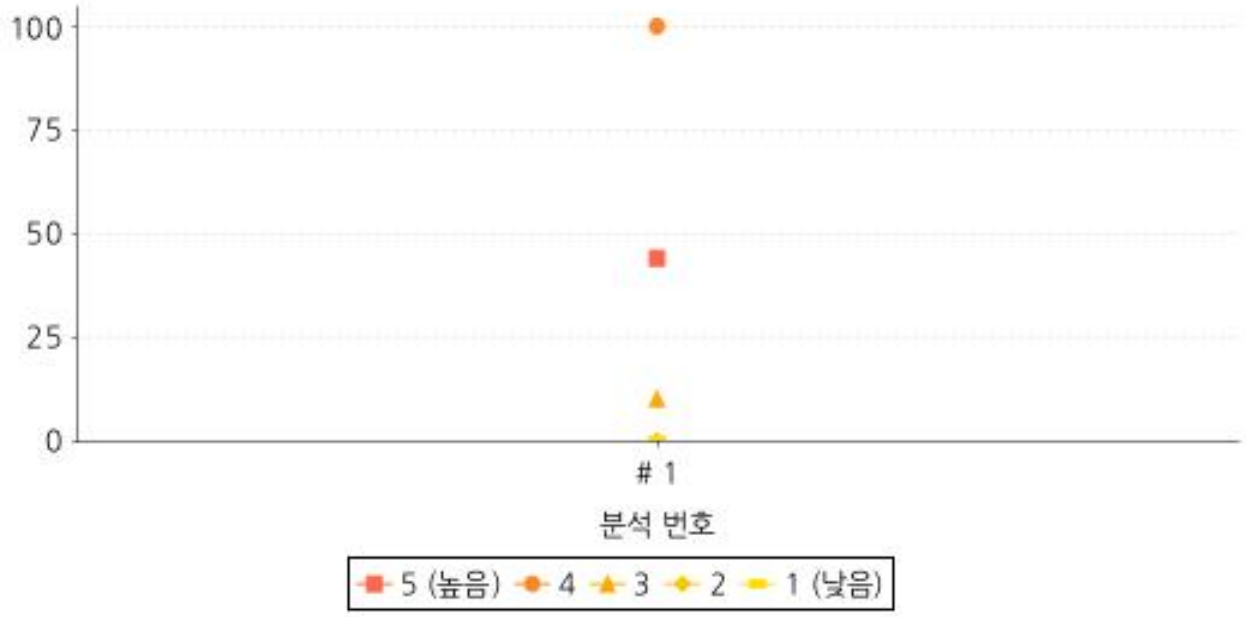
순위	규칙 이름	보안약점 수	위험도
1	부적절한 예외 처리	42	4
2	종료되지 않는 반복문 또는 재귀 함수	32	4
3	적절한 인증없는 중요 기능 허용	31	5
4	제거되지 않고 남은 디버그 코드	12	4
5	오류메시지를 통한 정보노출	7	3

2.4. 분석별 추이

분석별 보안약점 추이



분석별 위험도 추이



3. 보안 규칙별 결과 요약

3.1. 행안부 SW 보안약점 2021

순번	규칙 이름	위험도	CWE/ CERT	보안약점 개수	검출 파일수
1	SQL 삽입	5	CWE-89 IDS00-J	1	1
2	코드삽입	3	CWE-94	0	0
3	경로 조작 및 자원 삽입	5	CWE-99 CWE-22 FIO16-J IDS02-J	2	1
4	크로스사이트 스크립트	5	CWE-80 CWE-1336 CWE-79 IDS11-J IDS01-J	0	0
5	운영체제 명령어 삽입	5	CWE-78 IDS07-J	0	0
6	위험한 형식 파일 업로드	5	CWE-434	0	0
7	신뢰되지 않는 URL 주소로 자동 접속 연결	4	CWE-601	0	0
8	부적절한 XML 외부개체 참조	3	CWE-611	0	0
9	XML 삽입	3	CWE-643 CWE-652	0	0
10	LDAP 삽입	5	CWE-90	0	0
11	크로스사이트 요청 위조	5	CWE-352	0	0
12	서버사이드 요청 위조	3	CWE-918	0	0
13	HTTP 응답분할	5	CWE-113	2	2
14	정수형 오버플로우	4	CWE-190 NUM00-J	0	0
15	보안 기능 결정에 사용되는 부적절한 입력값	5	CWE-807	0	0
16	메모리 버퍼 오버플로우	5	CWE-119	0	0
17	포맷 스트링 삽입	5	CWE-134 IDS06-J	0	0

순번	규칙 이름	위험도	CWE/ CERT	보안약점 개수	검출 파일수
18	적절한 인증없는 중요 기능 허용	5	CWE-306	31	15
19	부적절한 인가	5	CWE-285	0	0
20	중요한 자원에 대한 잘못된 권한설정	4	CWE-732 ENV03-J FIO01-J	0	0
21	취약한 암호화 알고리즘 사용	4	CWE-327 MSC02-J	0	0
22	암호화 되지 않은 중요정보	3	CWE-312 CWE-319	0	0
23	하드코드 된 중요정보	3	CWE-321 CWE-259	0	0
24	충분하지 않은 키 길이 사용	4	CWE-326	0	0
25	적절하지 않은 난수 값 사용	5	CWE-330	4	4
26	취약한 비밀번호 허용	4	CWE-521	0	0
27	부적절한 전자서명 확인	3	CWE-347	0	0
28	부적절한 인증서 유효성 검증	3	CWE-295	0	0
29	사용자 하드디스크에 저장되는 쿠키를 통한 정보노출	4	CWE-539	0	0
30	주석문 안에 포함된 패스워드 등 시스템 주요정보	4	CWE-615	0	0
31	솔트 없이 일방향 해쉬 함수 사용	4	CWE-759	0	0
32	무결성 검사없는 코드 다운로드	4	CWE-494 FIO04-J	0	0
33	반복된 인증시도 제한 기능 부재	4	CWE-307	0	0
34	경쟁조건: 검사시점과 사용시점(TOCTOU)	4	CWE-367	0	0
35	종료되지 않는 반복문 또는 재귀 함수	4	CWE-674	32	6
36	오류 메시지 정보노출	3	CWE-209 CWE-497	7	5
37	오류상황 대응 부재	4	CWE-390 ERR00-J	4	3

순번	규칙 이름	위험도	CWE/ CERT	보안약점 개수	검출 파일수
38	부적절한 예외 처리	4	CWE-754 ERR06-J	42	16
39	널(Null) 포인터 역참조	5	CWE-476 EXP01-J	4	4
40	부적절한 자원 해제	4	CWE-404	5	3
41	해제된 자원 사용	4	CWE-416	0	0
42	초기화되지 않은 변수 사용	4	CWE-457	0	0
43	신뢰할 수 없는 데이터의 역질렬화	3	CWE-502	0	0
44	잘못된 세션에 의한 데이터 정보 노출	4	CWE-488	5	2
45	제거되지 않고 남은 디버그 코드	4	CWE-489	12	8
46	Public 메소드로부터 반환된 Private 배열	3	CWE-495	3	1
47	Private 배열에 Public 데이터 할당	3	CWE-496	0	0
48	DNS lookup 에 의존한 보안결정	4	CWE-247	0	0
49	취약한 API 사용	4	CWE-676	0	0

4. 보안약점 전체 목록

4.1. SQL 삽입 [0001_SQLI]

규칙 코드	0001_SQLI
CWE/CERT	CWE-89,IDS00-J
관련 규칙	행안부 SW 보안약점 2021 - SQL 삽입
규칙 설명	데이터베이스(DB)와 연동된 웹 애플리케이션에서 입력된 데이터에 대한 유효성 검증을 하지 않을 경우, 공격자가 입력 폼 및 URL 입력란에 SQL 문을 삽입하여 DB 로부터 정보를 열람하거나 조작할 수 있는 보안 약점을 말합니다. 취약한 웹 애플리케이션에서는 사용자로부터 입력된 값을 필터링 과정 없이 넘겨받아 동적 쿼리(Dynamic Query)를 생성하기 때문에 개발자가 의도하지 않은 쿼리가 생성되어 정보 유출에 악용될 수 있습니다.

올바른 예제

```
1  // JDBC API 관련 예
2  **String gubun = request.getParameter("gubun");
3  ……
4  // 1. 사용자에 의해 외부로부터 입력받은 값은 안전하지 않을 수 있으므로, PreparedStatement 사용을 위해 ?문자로 바인딩 변수를 사용합니다.
5  String sql = "SELECT * FROM board WHERE b_gubun = ?";
6  Connection con = db.getConnection();
7  // 2. PreparedStatement 사용합니다.
8  **PreparedStatement pstmt = con.prepareStatement(sql);
9  // 3. PreparedStatement 객체를 상수 스트링으로 생성하고, 파라미터 부분을 setString 등의 메소드로 설정하여 안전합니다.
10 **pstmt.setString(1, gubun);
11 **ResultSet rs = pstmt.executeQuery();
12
13 // MyBatis 관련 예
14 <select id="boardSearch" parameterType="map" resultType="BoardDto">
15 // $ 대신 #기호를 사용하여 변수가 쿼리맵에 바인딩 될 수 있도록 수정하는 것이 안전합니다.
16 **      select * from tbl_board where title like '%'||# {keyword }||'%' order by pos asc
17 </select>
```

파라미터(Parameter)를 받는 PreparedStatement 객체를 상수 스트링으로 생성하고, 파라미터 부분을 setXXX 메소드로 설정하여, 외부의 입력이 쿼리문의 구조를 바꾸는 것을 방지해야 합니다.

잘못된 예제

```
1  // JDBC API 관련 예
 2  //외부로부터 입력받은 값을 검증 없이 사용할 경우 안전하지 않습니다.
 3  **String gubun = request.getParameter("gubun");
 4  ......
 5  **String sql = "SELECT * FROM board WHERE b_gubun = '" + gubun + "'";
 6  Connection con = db.getConnection();
 7  **Statement stmt = con.createStatement();
 8  //외부로부터 입력받은 값이 검증 또는 처리 없이 쿼리로 수행되어 안전하지 않습니다.
 9  **ResultSet rs = stmt.executeQuery(sql);
10
11  // MyBatis 관련 예
12   < select id = "boardSearch" parameterType = "map" resultType = "BoardDto" >
13  // $기호를 사용하는 경우 외부에서 입력된 keyword 값을 문자열에 결합한 형태로 쿼리에
반영되므로 안전하지 않습니다.
14  **      select * from tbl_board where title like '%${keyword }%' order by pos asc
15   <  / select >
```

외부로부터 입력받은 gubun 의 값을 아무런 검증과정을 거치지 않고 SQL 쿼리를 생성하는데 사용하고 있습니다. 이 경우 gubun 의 값으로 a' or 'a' = 'a 를 입력하면 조건절이 b_gubun = 'a' or 'a' = 'a' 로 바뀌어 쿼리의 구조가 변경되어 board 테이블의 모든 내용이 조회됩니다.

순번	규칙 코드	위험도	규칙 이름
1	0001_SQLI	5	SQL 삽입
CWE/CERT		CWE-89,IDS00-J	
파일		/nmkpg_cyber_user/src/main/resources/egovframework/mapper/user/ExhibitMapper.xml	
발견 위치		43:30 - 43:43	

```
 40                    (SELECT code_name FROM st_code_dtl WHERE grp_code = 'LOCATION_TYPE'
AND code = A.location_type) AS location_type_name,
 41                    (SELECT code_name FROM st_code_dtl WHERE grp_code = 'ROAD_TYPE'
AND code = A.road_type) AS road_type_name
 42              FROM st_exhibit A
* 43              WHERE A.exhibit_seq = ${exhibitSeq}
 44          </select>
 45
 46     </mapper>
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

4.2. 경로 조작 및 자원 삽입 [0002_PATHMANIP]

규칙 코드	0002_PATHMANIP
CWE/CERT	CWE-99,CWE-22,FIO16-J,IDS02-J
관련 규칙	행안부 SW 보안약점 2021 - 경로 조작 및 자원 삽입
규칙 설명	검증되지 않은 외부 입력값을 통해 파일 및 서버 등 시스템 자원에 대한 접근 혹은 식별을 허용할 경우, 입력값 조작을 통해 시스템이 보호하는 자원에 임의로 접근할 수 있는 보안 약점입니다. 경로 조작 및 자원삽입 약점을 이용하여 공격자는 자원의 수정, 삭제, 시스템 정보누출, 시스템 자원 간 충돌로 인한 서비스 장애 등을 유발할 수 있습니다. 즉, 경로 조작 및 자원 삽입을 통해서 공격자가 허용되지 않은 권한을 획득하여, 설정에 관계된 파일을 변경하거나 실행시킬 수 있습니다.

올바른 예제

```
1  //예제 1
2   **String fileName = request.getParameter("P");
3   BufferedInputStream bis = null;
4   BufferedOutputStream bos = null;
5   FileInputStream fis = null;
6   try {
7         response.setHeader("Content-Disposition", "attachment;filename="+fileName+";");
8         ...
9         // 외부 입력받은 값을 경로순회 문자열(./₩)을 제거하고 사용해야 합니다.
10  **       filename = filename.replaceAll("₩₩.", "").replaceAll("/", "").replaceAll("₩₩₩₩", "");
11        fis = new FileInputStream("C:/datas/" + fileName);
12        bis = new BufferedInputStream(fis);
13        bos = new BufferedOutputStream(response.getOutputStream());
14        int read;
15        while((read = bis.read(buffer, 0, 1024)) != -1) {
16              bos.write(buffer,0,read);
17        }
18  }
19
20  //예제 2
21  public class ShowHelpSolution {
22        private final static String safeDir = "c:₩₩help_files₩₩";
23        //경로조작 문자열 포함 여부를 확인하고 조치 후 사용하도록 합니다.
```

```
24        public static void main(String[] args) throws IOException {
25  **      String helpFile = args[0];
26  **      if (helpFile != null) {
27  **          helpFile = helpFile.replaceAll("₩₩. {2, }[/₩₩₩₩]", "");
28        }
29  **      try (BufferedReader br = new BufferedReader(new FileReader(safeDir + helpFile))) {
30            ...
```

외부 입력값에 대하여 상대경로를 설정할 수 없도록 경로 순회 문자열(/ ₩ & .. 등)을 제거하고 파일의 경로 설정에 사용합니다.

잘못된 예제

```
1  //예제 1
2  //외부로부터 입력받은 값을 검증 없이 사용할 경우 안전하지 않습니다.
3  **String fileName = request.getParameter("P");
4  BufferedInputStream bis = null;
5  BufferedOutputStream bos = null;
6  FileInputStream fis = null;
7  try {
8        response.setHeader("Content-Disposition", "attachment;filename="+fileName+";");
9        ...
10       //외부로부터 입력받은 값이 검증 또는 처리 없이 파일처리에 수행되었습니다.
11 **     fis = new FileInputStream("C:/datas/" + fileName);
12       bis = new BufferedInputStream(fis);
13       bos = new BufferedOutputStream(response.getOutputStream());
14 }
15
16 //예제 2
17 public class ShowHelp {
18       private final static String safeDir = "c:₩₩help_files₩₩";
19       public static void main(String[] args) throws IOException {
20 **          String helpFile = args[0];
21 **          try (BufferedReader br = new BufferedReader(new FileReader(safeDir + helpFile))) {
22                String line;
23                while ((line = br.readLine()) != null) {
24                     System.out.println(line);
25                }
26                ...
```

```
27            }
28        }
29    }
```

외부 입력값(P)이 버퍼로 내용을 옮길 파일의 경로 설정에 사용되고 있습니다. 만일 공격자에 의해 P 의 값으로 ../../../rootFile.txt 와 같은 값을 전달하면 의도하지 않았던 파일의 내용이 버퍼에 쓰여 시스템에 악영향을 줍니다.

순번	규칙 코드	위험도	규칙 이름
2	0002_PATHMANIP	5	경로 조작 및 자원 삽입
CWE/CERT		CWE-99,CWE-22,FIO16-J,IDS02-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		347:14 - 347:32	

```
 344         }
  345
 346         public static File uploadFile(String filepath, MultipartFile source) throws IOException {
* 347             File file = new File(filepath);
  348             FileUtils.forceMkdirParent(file);
  349             FileUtils.copyInputStreamToFile(source.getInputStream(), file);
  350             return file;
```

코드 수정 제안	347 행 File(..) 함수 호출시에 외부로부터 온 값을 검증하지 않고 시스템 자원에 대한 식별자로 사용하는 경우, 공격자가 시스템이 보호하는 자원에 임의로 접근하거나 수정할 수 있습니다. 347 행 File(..) 함수 호출 에 대해 화이트 리스트 형태로 외부 입력을 여과하거나 경로에 접근할 경우에는 경로순회 문자열(/ ₩ & 등)을 제거하세요.

순번	규칙 코드	위험도	규칙 이름
3	0002_PATHMANIP	5	경로 조작 및 자원 삽입
CWE/CERT		CWE-99,CWE-22,FIO16-J,IDS02-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		354:14 - 354:32	

```
 351         }
  352
 353         public static void deleteFile(String filepath) throws IOException {
* 354             File file = new File(filepath);
```

```
355                 FileUtils.forceDelete(file);
356         }
```

코드 수정 제안	354 행 File(..) 함수 호출시에 외부로부터 온 값을 검증하지 않고 시스템 자원에 대한 식별자로 사용하는 경우, 공격자가 시스템이 보호하는 자원에 임의로 접근하거나 수정할 수 있습니다. 354 행 File(..) 함수 호출 에 대해 화이트 리스트 형태로 외부 입력을 여과하거나 경로에 접근할 경우에는 경로순회 문자열(/ ₩ & 등)을 제거하세요.

4.3. HTTP 응답분할 [000B_HTTPSPLIT]

규칙 코드	000B_HTTPSPLIT
CWE/CERT	CWE-113
관련 규칙	행안부 SW 보안약점 2021 - HTTP 응답분할
규칙 설명	HTTP 요청에 들어 있는 파라미터(Parameter)가 HTTP 응답 헤더에 포함되어 사용자에게 다시 전달될 때, 입력값에 CR(Carriage Return)이나 LF(Line Feed)와 같은 개행문자가 존재하면 HTTP 응답이 2개 이상으로 분리될 수 있습니다. 이 경우 공격자는 개행문자를 이용하여 첫 번째 응답을 종료시키고, 두 번째 응답에 악의적인 코드를 주입하여 XSS 및 캐시 훼손(Cache Poisoning) 공격 등을 수행할 수 있습니다.

올바른 예제

```
1   **String lastLogin = request.getParameter("last_login");
2   if (lastLogin == null || "".equals(lastLogin)) {
3         return;
4   }
5   // 외부 입력값에서 개행문자(₩r₩n)를 제거한 후 쿠키의 값으로 설정
6   **lastLogin = lastLogin.replaceAll("[₩₩r₩₩n]", "");
7   Cookie c = new Cookie("LASTLOGIN", lastLogin);
8   c.setMaxAge(1000);
9   c.setSecure(true);
10  response.addCookie(c);
```

외부에서 입력되는 값에 대하여 널(Null) 여부를 체크하고, 응답이 여러 개로 나눠지는 것을 방지하기 위해 개행문자를 제거하고 응답헤더의 값으로 사용합니다.

잘못된 예제

```
1  // 외부로부터 입력받은 값을 검증 없이 사용할 경우 안전하지 않습니다.
2  **String lastLogin = request.getParameter("last_login");
3  if (lastLogin == null || "".equals(lastLogin)) {
4        return;
5  }
6  // 쿠키는 Set-Cookie 응답헤더로 전달되므로 개행문자열 포함 여부 검증이 필요
7  **Cookie c = new Cookie("LASTLOGIN", lastLogin);
8  c.setMaxAge(1000);
9  c.setSecure(true);
10 **response.addCookie(c);
11 response.setContentType("text/html");
```

외부 입력값을 사용하여 반환되는 쿠키의 값을 설정하고 있습니다. 그런데, 공격자가 Wiley Hacker₩r₩nHTTP/1.1 200 OK₩r₩n 를 lastLogin 의 값으로 설정할 경우, 응답이 분리되어 전달되며 분리된 응답 본문의 내용을 공격자가 마음대로 수정할 수 있습니다.

순번	규칙 코드	위험도	규칙 이름
4	000B_HTTPSPLIT	5	HTTP 응답분할
CWE/CERT		CWE-113	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		376:3 - 376:95	

```
373                 }
374
375                 String encodeFilename = encodeFilename(filename, request);
* 376                 response.setHeader("Content-Disposition", "attachment; filename=₩"" +
encodeFilename + "₩"");
377                 response.setContentLength((int) file.length());
378
379                 try (InputStream inputStream =
Files.newInputStream(Paths.get(file.getAbsolutePath()))) {
```

코드 수정 제안	376 행 setHeader(..) 함수 호출시에 395 행 getHeader(..) 함수 호출 부터 온 값에 CR(Carriage Return)이나 LF(Line Feed)와 같은 개행문자가 존재하면 HTTP 응답이 2개로 분리될 수 있습니다. 395 행 getHeader(..) 함수 호출 값을 아래와 같이 치환함으로써 헤더값이 둘로 나뉘는 것을 방지할 수 있습니다. ... = request.getHeader(..);

	...
	author = ...;
	// 개행문자를 제거하여 헤더값이 둘로 나뉘는 것을 방지한다.
	String filtered_author = author.replaceAll("₩r", "").replaceAll("₩n", "");
	Cookie cookie = new Cookie(""replidedAuthor"", filtered_author);
	cookie.setMaxAge(1000);
	response.addCookie(cookie);

순번	규칙 코드	위험도	규칙 이름
5	000B_HTTPSPLIT	5	HTTP 응답분할
CWE/CERT		CWE-113	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		279:2 - 279:87	

```
 276
 277                response.setContentType("application/vnd.ms-excel");
  278                response.setCharacterEncoding("UTF-8");
* 279                response.setHeader("Content-Disposition", "attachment;filename=" + filename + ".xls");
  280                response.setHeader("Cache-Control", "max-age=0");
  281
 282                OutputStream os = null;
```

코드 수정 제안	279 행 setHeader(..) 함수 호출시에 395 행 getHeader(..) 함수 호출 부터 온 값에 CR(Carriage Return)이나 LF(Line Feed)와 같은 개행문자가 존재하면 HTTP 응답이 2개로 분리될 수 있습니다.
	
	395 행 getHeader(..) 함수 호출 값을 아래와 같이 치환함으로써 헤더값이 둘로 나뉘는 것을 방지할 수 있습니다.
	... = request.getHeader(..);
	...
	author = ...;
	// 개행문자를 제거하여 헤더값이 둘로 나뉘는 것을 방지한다.
	String filtered_author = author.replaceAll("₩r", "").replaceAll("₩n", "");
	Cookie cookie = new Cookie(""replidedAuthor"", filtered_author);
	cookie.setMaxAge(1000);
	response.addCookie(cookie);

4.4. 적절한 인증없는 중요 기능 허용 [0010_MISSAUTH]

규칙 코드	0010_MISSAUTH
CWE/CERT	CWE-306,CWE-287
관련 규칙	행안부 SW 보안약점 2021 - 적절한 인증없는 중요 기능 허용
규칙 설명	적절한 인증 과정 없이 중요정보(계좌이체 정보, 개인정보 등)를 열람(또는 변경)할 때 발생하는 보안 약점입니다.

올바른 예제

```
1   @RequestMapping(value = "/modify.do", method = RequestMethod.POST)
 2   public ModelAndView memberModifyProcess(@ModelAttribute("MemberModel") MemberModel
memberModel, BindingResult result, HttpServletRequest request, HttpSession session) {
 3         ModelAndView mav = new ModelAndView();
 4
5         //1. 로그인한 사용자를 불러옵니다.
 6   **       String userId = (String) session.getAttribute("userId");
 7         String passwd = request.getParameter("oldUserPw");
 8
9         //2. 회원정보를 실제 수정하는 사용자와 로그인 사용자와 동일한지 확인합니다.
10   **       String requestUser = memberModel.getUserId();
11   **       if (userId != null &amp;amp;&amp;amp; requestUser != null &amp;amp;&amp;amp;
!userId.equals(requestUser)) {
12               mav.addObject("errCode", 1);
13               mav.addObject("member", memberModel);
14               mav.setViewName("/board/member_modify");
15               return mav;
16         }
17         ...
18
19         //3. 동일한 경우에만 회원정보를 수정해야 안전합니다.
20   **       if (service.modifyMember(memberModel)) {
21               ...
```

로그인한 사용자와 요청한 사용자의 일치 여부를 확인한 후 회원 정보를 수정하도록 합니다.

잘못된 예제

```
1   @RequestMapping(value = "/modify.do", method = RequestMethod.POST)
```

```
 2   public ModelAndView memberModifyProcess(@ModelAttribute("MemberModel") MemberModel
memberModel, BindingResult result, HttpServletRequest request, HttpSession session) {
 3   ModelAndView mav = new ModelAndView();
 4         //1. 로그인한 사용자를 불러옵니다.
 5   **      String userId = (String) session.getAttribute("userId");
 6         String passwd = request.getParameter("oldUserPw");
 7         ...
 8         //2. 실제 수정하는 사용자와 일치 여부를 확인하지 않고, 회원정보를 수정하여 안전하지
않습니다.
 9   **      if (service.modifyMember(memberModel)) {
10               mav.setViewName("redirect:/board/list.do");
11               session.setAttribute("userName", memberModel.getUserName());
12               return mav;
13         } else {
14               mav.addObject("errCode", 2);
15               mav.setViewName("/board/member_modify");
16               return mav;
17         }
18   }
```

회원 정보 수정 시 수정을 요청한 사용자와 로그인한 사용자의 일치 여부를 확인하지 않고 처리하고 있습니다.

순번	규칙 코드	위험도	규칙 이름
6	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/component/PasswordConstraintValidator.java	
발견 위치		40:32 - 40:59	

```
 37
 38          @Override
  39          public boolean isValid(String password, ConstraintValidatorContext context) {
* 40               PasswordValidator validator = this.getPasswordVAlidator();
  41
 42               RuleResult result = this.validate(password);
  43               if (result.isValid()) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
7	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/component/PasswordConstraintValidator.java	
발견 위치		54:32 - 54:59	

```
 51          }
 52
 53          public RuleResult validate(String password) {
* 54                  PasswordValidator validator = this.getPasswordVAlidator();
 55                   RuleResult result = validator.validate(new PasswordData(password));
 56                   return result;
 57           }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
8	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/BaseController.java	
발견 위치		108:9 - 108:29	

```
 105
 106
 107          public List<Map<String, Object>> getAccountList() {
* 108                  return getAccountList(true);
 109           }
 110
 111          public List<Map<String, Object>> getAccountList(boolean addEmpty) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
9	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/BaseController.java	
발견 위치		112:35 - 112:66	

```
 109         }
  110
 111         public List<Map<String, Object>> getAccountList(boolean addEmpty) {
* 112             List<Map<String, Object>> list = accountService.getAccountCode();
  113             if (addEmpty) {
  114                 Map<String, Object> map = new HashMap<String, Object>();
  115                 map.put("code", 0);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
10	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/SettingController.java	
발견 위치		169:54 - 169:72	

```
 166                 if (StringUtil.isBlank(user.getPassword())) {
  167                     bindingResult.rejectValue("password", "field.required");
  168                 }
* 169                 RuleResult ruleResult = passwordValidator.validate(user.getPassword());
  170                 if (!ruleResult.isValid()) {
  171                     bindingResult.rejectValue("password", "password.illegal_match");
  172                 }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
11	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/SettingController.java	
발견 위치		474:35 - 474:83	

```
 471
 472             Map<String, Object> paramsMap = StoneUtil.convertObjectToMap(params);
  473
* 474             List<Map<String, Object>> list = accountService.getAccountList(paramsMap, paging);
  475
 476             model.addAttribute("list", list);
```

```
477                 model.addAttribute("bankList", getCodeList("BANK"));
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
12	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/SettingController.java	
발견 위치		510:21 - 510:97	

```
507         @ResponseBody
508         public Map<String, Object> getAccount(@RequestBody Map<String, Object> params) {
509             Map<String, Object> result = new HashMap<String, Object>();
* 510             result.put("data",
accountService.getAccount(StringUtil.getIntValue(params.get("account_seq"))));
511             result.put("result_code", GlobalConstant.API_STATUS.SUCCESS);
512             return result;
513         }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
13	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/security/AuthProvider.java	
발견 위치		64:58 - 64:76	

```
61                     throw new BadCredentialsException(id);
62                 }
63             } else {
* 64                 if (user == null || !passwordEncoder.matches(password, user.getPassword())) {
65                     throw new BadCredentialsException(id);
66                 }
67             }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름

14	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/services/AccountService.java	
발견 위치		21:9 - 21:45	

```
 18         private AccountMapper accountMapper;
 19
 20         public List<Map<String, Object>> getAccountList(Map<String, Object> params) {
 * 21             return accountMapper.getAccountList(params);
 22         }
 23
 24         public List<Map<String, Object>> getAccountList(Map<String, Object> params, Paging
paging) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
15	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/services/AccountService.java	
발견 위치		25:9 - 25:65	

```
 22         }
 23
 24         public List<Map<String, Object>> getAccountList(Map<String, Object> params, Paging
paging) {
 * 25             return accountMapper.getAccountList(params, paging.getPaging());
 26         }
 27
 28         public int existedAccount(Account account) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
16	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/services/AccountService.java	

발견 위치	51:9 - 51:45

```
 48          }
  49
 50          public Map<String, Object> getAccount(int accountSeq) {
* 51                return accountMapper.getAccount(accountSeq);
  52           }
  53
 54          public List<Map<String, Object>> getAccountCode() {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
17	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/services/AccountService.java	
발견 위치		55:9 - 55:39	

```
 52          }
  53
 54          public List<Map<String, Object>> getAccountCode() {
* 55                return accountMapper.getAccountCode();
  56           }
  57
 58     }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
18	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/services/auth/UserService.java	
발견 위치		87:42 - 87:60	

```
 84
 85          public int saveUser(User user) {
  86               if (StringUtil.isNotBlank(user.getPassword())) {
* 87                    user.setUserPwd(passwordEncoder.encode(user.getPassword()));
  88               }
```

```
 89
 90                 return userMapper.insert(user);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
19	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/bootstrap-datepicker.min.js	
발견 위치		7:18031 - 7:18051	

```
 4      * Licensed under the Apache License v2.0 (http://www.apache.org/licenses/LICENSE-2.0)
 5      */
 6
* 7     ... ullYear(),0,1))+(11-C.getUTCDay())%7*864e5),D=(B-C)/864e5/7+1;z.push('<td
class="cw">'+D+"</td>")}y=this.getClassNames(u),y.push("day");var
E=u.getUTCDate();this.o.beforeShowDay!==a.noop&&(f=this.o.beforeShowDay(this._utc ...
 8      headTemplate:'<thead><tr><th colspan="7" class="datepicker-title"></th></tr><tr><th
class="prev">'+o.templates.leftArrow+'</th><th colspan="5" class="datepicker-switch"></th><th
class="next">'+o.templates.rightArrow+"</th></tr></thead>",contTemplate:'<tbody><tr><td
colspan="7"></td></tr></tbody>',footTemplate:'<tfoot><tr><th colspan="7"
class="today"></th></tr><tr><th colspan="7" class="clear"></th></tr></tfoot>'};r.template='<div
class="datepicker"><div class="datepicker-days"><table
class="table-condensed">'+r.headTemplate+"<tbody></tbody>"+r.footTemplate+'</table></div><di
v class="datepicker-months"><table
class="table-condensed">'+r.headTemplate+r.contTemplate+r.footTemplate+'</table></div><div
class="datepicker-years"><table
class="table-condensed">'+r.headTemplate+r.contTemplate+r.footTemplate+'</table></div><div
class="datepicker-decades"><table
class="table-condensed">'+r.headTemplate+r.contTemplate+r.footTemplate+'</table></div><div
class="datepicker-centuries"><table
class="table-condensed">'+r.headTemplate+r.contTemplate+r.footTemplate+"</table></div></div>",
a.fn.datepicker.DPGlobal=r,a.fn.datepicker.noConflict=function(){return
a.fn.datepicker=m,this},a.fn.datepicker.version="1.9.0",a.fn.datepicker.deprecated=function(a){var
b=window.console;b&&b.warn&&b.warn("DEPRECATED:
"+a)},a(document).on("focus.datepicker.data-api
click.datepicker.data-api",'[data-provide="datepicker"]',function(b){var
c=a(this);c.data("datepicker")||(b.preventDefault(),n.call(c,"show"))}),a(function(){n.call(a('[data-provide="d
atepicker-inline"]'))})});
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
20	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/pages/setting/account_list.js	
발견 위치		33:16 - 33:37	

```
 30
 31              function openMenuModal(accountSeq) {
  32                  if (accountSeq) {
* 33                      getAccount(accountSeq);
  34                  } else {
  35                      $('#account_seq').val('');
  36                      $('#bank').val('');
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
21	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		215:23 - 216:57	

```
 212
 213                                 if (cols[0][j].isMoney()) {
  214                                     try {
* 215                                         formatString =
StringUtil.getAccountFormat(cells.getValue("ExcelSymbol"),
  216                                                 Integer.parseInt(cells.getValue("Precisions")));
  217                                     } catch (Exception e) {
  218                                         formatString = "#,##0";
  219                                     }
```

순번	규칙 코드	위험도	규칙 이름

22	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/component/PasswordConstraintValidator.java	
발견 위치		40:32 - 40:59	

```
 37
 38         @Override
  39         public boolean isValid(String password, ConstraintValidatorContext context) {
* 40             PasswordValidator validator = this.getPasswordVAlidator();
  41
 42             RuleResult result = this.validate(password);
  43             if (result.isValid()) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
23	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/component/PasswordConstraintValidator.java	
발견 위치		54:32 - 54:59	

```
 51         }
  52
 53         public RuleResult validate(String password) {
* 54             PasswordValidator validator = this.getPasswordVAlidator();
  55             RuleResult result = validator.validate(new PasswordData(password));
  56             return result;
  57         }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
24	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AuthenticationController.java	
발견 위치		128:81 - 128:99	

```
125
```

```
  126                     User user = userService.findByUserId(userIdFromRefreshToken);
  127
* 128                   if (user != null && StringUtils.equals(passwordFromRefreshToken,
user.getPassword())
  129                         && StringUtils.equals(refreshToken, user.getRefreshToken())) {
  130                   String jwtToken = tokenProvider.generateToken(user);
  131                   Date expiration = tokenProvider.getTokenExpiration(jwtToken);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
25	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AuthenticationController.java	
발견 위치		249:83 - 249:107	

```
  246               String userId = RequestUtils.replaceEsSpecialChars(form.getUser_id());
  247                User searchUser = userService.findByUserId(userId);
  248
* 249               if (searchUser == null || !passwordEncoder.matches(form.getUser_pwd(),
searchUser.getPassword())) {
  250                   CommonErrorResponse commonResponse = new CommonErrorResponse();
  251                   commonResponse.setStatus(400);
  252                   commonResponse.setErrorCode("login.error.wrongIdPassword");
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
26	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		289:31 - 289:58	

```
  286         if (isValidDate(form.getSchedule_date(), "yyyyMMddHHmmss") &&
isValidDate(form.getSchedule_date(), "yyyyMMddHHmm")) {
  287             return inputFieldBadRequest("schedule_date", form.getSchedule_date());
  288         }
 * 289         if (!StringUtils.equalsAny(form.getPassword_use_flag(), "0", "1")) {
```

```
290             return inputFieldBadRequest("password_use_flag", form.getPassword_use_flag());
291         }
292         if (!StringUtils.equalsAny(form.getAdd_join_flag(), "0", "1")) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
27	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		290:55 - 290:82	

```
287             return inputFieldBadRequest("schedule_date", form.getSchedule_date());
288         }
289         if (!StringUtils.equalsAny(form.getPassword_use_flag(), "0", "1")) {
* 290           return inputFieldBadRequest("password_use_flag", form.getPassword_use_flag());
291         }
292         if (!StringUtils.equalsAny(form.getAdd_join_flag(), "0", "1")) {
293             return inputFieldBadRequest("add_join_flag", form.getAdd_join_flag());
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
28	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		382:57 - 382:84	

```
379         } else {
380             data.put("schedule_date", null);
381         }
* 382       data.put("password_use_flag", BooleanUtils.toInteger(docent.getPasswordUseFlag()));
383         data.put("use_password", docent.getUsePassword());
384         data.put("docent_desc", docent.getDocentDesc());
385         data.put("member_count", docent.getMemberCount());
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
29	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		383:29 - 383:52	

```
380             data.put("schedule_date", null);
381         }
382         data.put("password_use_flag", BooleanUtils.toInteger(docent.getPasswordUseFlag()));
* 383       data.put("use_password", docent.getUsePassword());
384         data.put("docent_desc", docent.getDocentDesc());
385         data.put("member_count", docent.getMemberCount());
386         data.put("add_join_flag", BooleanUtils.toInteger(docent.getAddJoinFlag()));
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
30	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/security/AuthProvider.java	
발견 위치		48:61 - 48:83	

```
45                  User authUser = authenticateUser(username, password);
46
47                  if (authUser != null) {
* 48                     return new UsernamePasswordAuthenticationToken(authUser,
authUser.getPassword(), authUser.getAuthorities());
49                  } else {
50                      return null;
51                  }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
31	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/security/AuthProvider.java	

발견 위치	74:41 - 74:63

```
 71                     throw new BadCredentialsException(INVALID_USERNAME_OR_PASSWORD);
 72                 }
 73
* 74                 if (!passwordEncoder.matches(password, authUser.getPassword())) {
 75                     throw new BadCredentialsException(INVALID_USERNAME_OR_PASSWORD);
 76                 }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
32	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/services/DocentService.java	
발견 위치		39:41 - 39:68	

```
 36         docent.setDocentSeq(Long.parseLong(form.getDocent_seq()));
 37         docent.setDocentTitle(form.getDocent_title());
 38         docent.setScheduleDate(form.getSchedule_date());
* 39         docent.setPasswordUseFlag("1".equals(form.getPassword_use_flag()));
 40         docent.setUsePassword(form.getUse_password());
 41         docent.setDocentDesc(form.getDocent_desc());
 42         docent.setMemberCount(Long.parseLong(form.getMember_count()));
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
33	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/services/DocentService.java	
발견 위치		40:26 - 40:48	

```
 37         docent.setDocentTitle(form.getDocent_title());
 38         docent.setScheduleDate(form.getSchedule_date());
 39         docent.setPasswordUseFlag("1".equals(form.getPassword_use_flag()));
* 40         docent.setUsePassword(form.getUse_password());
 41         docent.setDocentDesc(form.getDocent_desc());
 42         docent.setMemberCount(Long.parseLong(form.getMember_count()));
```

43 docent.setAddJoinFlag("1".equals(form.getAdd_join_flag()));	
코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
34	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/services/DocentService.java	
발견 위치		55:41 - 55:68	
52 Docent docent = new Docent(); 53 docent.setDocentTitle(form.getDocent_title()); 54 docent.setScheduleDate(form.getSchedule_date()); * 55 docent.setPasswordUseFlag("1".equals(form.getPassword_use_flag())); 56 docent.setUsePassword(form.getUse_password()); 57 docent.setDocentDesc(form.getDocent_desc()); 58 docent.setMemberCount(Long.parseLong(form.getMember_count()));			
코드 수정 제안		적합한 수정제안이 존재하지 않습니다.	

순번	규칙 코드	위험도	규칙 이름
35	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용
CWE/CERT		CWE-306,CWE-287	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/services/DocentService.java	
발견 위치		56:26 - 56:48	
53 docent.setDocentTitle(form.getDocent_title()); 54 docent.setScheduleDate(form.getSchedule_date()); 55 docent.setPasswordUseFlag("1".equals(form.getPassword_use_flag())); * 56 docent.setUsePassword(form.getUse_password()); 57 docent.setDocentDesc(form.getDocent_desc()); 58 docent.setMemberCount(Long.parseLong(form.getMember_count())); 59 docent.setAddJoinFlag("1".equals(form.getAdd_join_flag()));			
코드 수정 제안		적합한 수정제안이 존재하지 않습니다.	

순번	규칙 코드	위험도	규칙 이름
36	0010_MISSAUTH	5	적절한 인증없는 중요 기능 허용

CWE/CERT	CWE-306,CWE-287
파일	/nmkpg_cyber_user/src/main/java/egovframework/iam/user/services/auth/AuthenticationService.java
발견 위치	44:28 - 44:46

```
 41                 authentication = authenticationManager.authenticate(
 42                         new UsernamePasswordAuthenticationToken(
 43                                 form.getUserId(),
* 44                                 form.getPassword()
 45                         )
 46                 );
 47             } catch (BadCredentialsException | LockedException e) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

4.5. 적절하지 않은 난수 값 사용 [0018_RANDOM]

규칙 코드	0018_RANDOM
CWE/CERT	CWE-330
관련 규칙	행안부 SW 보안약점 2021 - 적절하지 않은 난수 값 사용
규칙 설명	예측 가능한 난수를 사용하는 것은 시스템에 보안 약점을 야기합니다. 예측 불가능한 숫자가 필요한 상황에서 예측 가능한 난수를 사용한다면, 공격자는 SW 에서 생성되는 다음 숫자를 예상하여 시스템을 공격하는 것이 가능합니다.

올바른 예제

```
1   import java.util.Random;
 2   **import java.security.SecureRandom;
 3   ...
 4   public Static int getRandomValue(int maxValue) {
 5         // setSeed 로 매번 변경되는 시드값을 설정 하거나, 기본값인 현재 시간 기반으로 매번 변경되는 시드값을 사용하도록 합니다.
 6   **       Random random = new Random();
 7   **       return random.nextInt(maxValue);
 8   }
 9   public Static String getAuthKey() {
10         // 보안결정을 위한 난수로는 예측이 거의 불가능하게 암호학적으로 보호된 SecureRandom 을 사용합니다.
11         try {
12   **            SecureRandom secureRandom = SecureRandom.getInstance("SHA1PRNG");
13   **            MessageDigest digest = MessageDigest.getInstance("SHA-256");
```

```
14  **          secureRandom.setSeed(secureRandom.generateSeed(128));
15  **          String authKey = new String(digest.digest((secureRandom.nextLong() + "").getBytes()));
16              ...
17      } catch (NoSuchAlgorithmException e) {
```

보안 결정을 위해 난수 사용 시에는 java.security.SecureRandom 클래스를 사용하는 것이 더 안전합니다.

잘못된 예제

```
1   import java.util.Random;
2   ...
3   public Static int getRandomValue(int maxValue){
4       // 고정된 시드값을 사용하여 동일한 난수값이 생성되어 안전하지 않습니다.
5   **      Random random=new Random(100);
6   **      return random.nextInt(maxValue);
7   }
8   public Static String getAuthKey(){
9       // 매번 변경되는 시드값을 사용하여 다른 난수값이 생성되나 보안결정을 위한 난수로는 안전하지 않습니다.
10  **      Random random=new Random();
11  **      String authKey=Integer.toString(random.nextInt());
```

java.util.Random 클래스의 random() 메소드 사용시, 고정된 seed 를 설정하면 동일한 난수 값이 생성되어 안전하지 않습니다. 매번 변경되는 seed 를 설정하더라도 보안결정을 위한 난수 이용시에는 안전하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
37	0018_RANDOM	5	적절하지 않은 난수 값 사용
CWE/CERT		CWE-330	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/bootstrap.js	
발견 위치		151:21 - 151:33	

```
148         getUID: function getUID(prefix) {
149             do {
150                 // eslint-disable-next-line no-bitwise
* 151             prefix += ~~(Math.random() * MAX_UID); // "~~" acts like a faster Math.floor() here
152             } while (document.getElementById(prefix));
153
154             return prefix;
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
38	0018_RANDOM	5	적절하지 않은 난수 값 사용
CWE/CERT		CWE-330	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/bootstrap4-dialog/js/bootstrap-dialog.js	
발견 위치		1277:20 - 1277:32	

```
1274          */
 1275         BootstrapDialog.newGuid = function () {
 1276             return 'xxxxxxxx-xxxx-4xxx-yxxx-xxxxxxxxxxxx'.replace(/[xy]/g, function (c) {
* 1277                 var r = Math.random() * 16 | 0, v = c === 'x' ? r : (r & 0x3 | 0x8);
 1278                 return v.toString(16);
 1279             });
 1280         };
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
39	0018_RANDOM	5	적절하지 않은 난수 값 사용
CWE/CERT		CWE-330	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/vendors/feather-icons/feather.js	
발견 위치		1939:14 - 1939:26	

```
1936  /***/ (function(module, exports) {
 1937
 1938  var id = 0;
* 1939   var postfix = Math.random();
 1940
 1941  module.exports = function (key) {
 1942      return 'Symbol('.concat(key === undefined ? '' : key, ')_', (++id + postfix).toString(36));
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
40	0018_RANDOM	5	적절하지 않은 난수 값 사용
CWE/CERT		CWE-330	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/StoneUtil.java	

발견 위치	200:16 - 200:29

```
 197         }
  198
 199         public static int randomRange(int n1, int n2) {
* 200             return (int) (Math.random() * (n2 - n1 + 1)) + n1;
  201         }
  202
 203         public static String getKoreanDateFormat(String strDate) {
```

코드 수정 제안	200행 Math.random 은 예측가능한 난수 입니다. 200행 Math.random 대신 java.security.SecureRandom 을 사용합니다.

4.6. 종료되지 않는 반복문 또는 재귀 함수 [0021_NONTERM]

규칙 코드	0021_NONTERM
CWE/CERT	CWE-674
관련 규칙	행안부 SW 보안약점 2021 - 종료되지 않는 반복문 또는 재귀 함수
규칙 설명	재귀의 순환 횟수를 제어하지 못하여 할당된 메모리나 프로그램 스택 등의 자원을 과다하게 사용하면 위험합니다. 대부분의 경우, 귀납 조건(Base Case)이 없는 재귀함수는 무한 루프에 빠져들게 되고 자원 고갈을 유발함으로써 시스템의 정상적인 서비스를 제공할 수 없게 합니다.

올바른 예제

```
1   public class SafeFactorial {
 2       public int factorial(int n) {
 3           int i;
 4       // 제어문을 통해 루프를 빠져나올 수 있게 기술되어야 합니다.
 5   **      if (n==1) {
 6   **          i=1;
 7   **      } else {
 8   **          i = n * factorial(n-1);
 9       }
10       return i;
11       }
12   }
```

제어문을 통해 루프를 빠져나올 수 있게 기술되어야 합니다.

잘못된 예제

```
1   public class UnsafeFactorial {
 2        // 적절한 제어문 없이 재귀호출하여 무한재귀가 됩니다.
 3        public int factorial(int n) {
 4  **            return n * factorial(n-1);
 5        }
 6   }
```

적절한 제어문 없이 재귀호출 하여 무한재귀가 됩니다.

순번	규칙 코드	위험도	규칙 이름
41	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/app-script.js	
발견 위치		49:12 - 61:12	

```
 46
 47          // === sidebar menu activation js
  48          $(function() {
* 49                  for (var i = window.location, o = $(".metismenu li a").filter(function() {
  50                          // const urlParams = new URLSearchParams(i.search);
  51                          // let menuCode = urlParams.get('menuCode');
  52                          let menuCode = $.urlParam('menuCode');
  53                          if (!menuCode) {
  54                                menuCode = 'M00';
  55                          }
  56                          return menuCode == this.dataset['menuCode'];
  57                          // return this.href == i;
  58                      }).addClass("").parent().addClass("mm-active");;) {
  59                      if (!o.is("li")) break;
  60                      o = o.parent("").addClass("mm-show").parent("").addClass("mm-active");
  61                  }
  62              }),
  63              /* Back To Top */
  64              $(document).ready(function() {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
42	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/codemirror.js	
발견 위치		1970:6 - 1984:6	

```
 1967            if (pos >= len) { break }
  1968
  1969            var upto = Math.min(len, nextChange);
* 1970            while (true) {
  1971              if (text) {
  1972                var end = pos + text.length;
  1973                if (!collapsed) {
  1974                  var tokenText = end > upto ? text.slice(0, upto - pos) : text;
  1975                  builder.addToken(builder, tokenText, style ? style + spanStyle : spanStyle,
  1976                                   spanStartStyle, pos + tokenText.length == nextChange ?
spanEndStyle : "", css, attributes);
  1977                }
  1978                if (end >= upto) {text = text.slice(upto - pos); pos = upto; break}
  1979                pos = end;
  1980                spanStartStyle = "";
  1981              }
  1982              text = allText.slice(at, at = styles[i++]);
  1983              style = interpretTokenStyle(styles[i++], builder.cm.options);
  1984            }
  1985          }
  1986        }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
43	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/codemirror.js	
발견 위치		5309:4 - 5322:4	

```
 5306          if (i == source.length) { return }
  5307          hist.lastOrigin = hist.lastSelOrigin = null;
  5308
* 5309          for (;;) {
  5310            event = source.pop();
```

```
5311            if (event.ranges) {
5312              pushSelectionToHistory(event, dest);
5313              if (allowSelectionOnly && !event.equals(doc.sel)) {
5314                setSelection(doc, event, {clearRedo: false});
5315                return
5316              }
5317              selAfter = event;
5318            } else if (suppress) {
5319              source.push(event);
5320              return
5321            } else { break }
5322          }
5323
5324        // Build up a reverse change object to add to the opposite history
5325         // stack (redo when undoing, and vice versa).
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
44	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/codemirror.js	
발견 위치		8311:15 - 8316:8	

```
8308            var before = 0, after = (styles.length - 1) / 2, ch = pos.ch;
8309              var type;
8310              if (ch == 0) { type = styles[2]; }
* 8311            else { for (;;) {
8312                var mid = (before + after) >> 1;
8313                if ((mid ? styles[mid * 2 - 1] : 0) >= ch) { after = mid; }
8314                else if (styles[mid * 2 + 1] < ch) { before = mid + 1; }
8315                else { type = styles[mid * 2 + 2]; break }
8316              } }
8317              var cut = type ? type.indexOf("overlay ") : -1;
8318              return cut < 0 ? type : cut == 0 ? null : type.slice(0, cut - 1)
8319            },
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름

45	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		3420:9 - 3420:21	

```
 3417    ;// CONCATENATED MODULE: ../../node_modules/prosemirror-model/dist/index.js
  3418
 3419
* 3420    function findDiffStart(a, b, pos) {
  3421        for (let i = 0;; i++) {
  3422            if (i == a.childCount || i == b.childCount)
  3423                return a.childCount == b.childCount ? null : pos;
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
46	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		3444:9 - 3444:19	

```
 3441            pos += childA.nodeSize;
  3442        }
  3443    }
* 3444    function findDiffEnd(a, b, posA, posB) {
  3445        for (let iA = a.childCount, iB = b.childCount;;) {
  3446            if (iA == 0 || iB == 0)
  3447                return iA == iB ? null : { a: posA, b: posB };
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
47	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		5214:4 - 5225:4	

```
 5211    }
```

```
  5212    function parseExprSubscript(stream) {
  5213        let expr = parseExprAtom(stream);
* 5214        for (;;) {
  5215            if (stream.eat("+"))
  5216                expr = { type: "plus", expr };
  5217            else if (stream.eat("*"))
  5218                expr = { type: "star", expr };
  5219            else if (stream.eat("?"))
  5220                expr = { type: "opt", expr };
  5221            else if (stream.eat("{"))
  5222                expr = parseExprRange(stream, expr);
  5223            else
  5224                break;
  5225        }
  5226        return expr;
  5227    }
  5228    function parseNum(stream) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
48	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		5311:12 - 5316:12	

```
  5308                return expr.exprs.reduce((out, expr) => out.concat(compile(expr, from)), []);
  5309            }
  5310            else if (expr.type == "seq") {
* 5311                for (let i = 0;; i++) {
  5312                    let next = compile(expr.exprs[i], from);
  5313                    if (i == expr.exprs.length - 1)
  5314                        return next;
  5315                    connect(next, from = node());
  5316                }
  5317            }
  5318            else if (expr.type == "star") {
  5319                let loop = node();
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
49	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		5387:13 - 5387:19	

```
 5384    function dfa(nfa) {
  5385        let labeled = Object.create(null);
  5386        return explore(nullFrom(nfa, 0));
* 5387        function explore(states) {
  5388            let out = [];
  5389            states.forEach(node => {
  5390                nfa[node].forEach(({ term, to }) => {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
50	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		7471:9 - 7471:19	

```
 7468        }
  7469    }
  7470
* 7471    function mapFragment(fragment, f, parent) {
  7472        let mapped = [];
  7473        for (let i = 0; i < fragment.childCount; i++) {
  7474            let child = fragment.child(i);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
51	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor	

	-all.js
발견 위치	7921:4 - 7928:4

```
 7918    function dist_liftTarget(range) {
  7919        let parent = range.parent;
  7920        let content = parent.content.cutByIndex(range.startIndex, range.endIndex);
* 7921        for (let depth = range.depth;; --depth) {
  7922            let node = range.$from.node(depth);
  7923            let index = range.$from.index(depth), endIndex = range.$to.indexAfter(depth);
  7924            if (depth < range.depth && node.canReplace(index, endIndex, content))
  7925                return depth;
  7926            if (depth == 0 || node.type.spec.isolating || !canCut(node, index, endIndex))
  7927                break;
  7928        }
  7929        return null;
  7930    }
  7931    function lift(tr, range, target) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
52	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		8098:4 - 8119:4	

```
 8095    */
  8096    function dist_joinPoint(doc, pos, dir = -1) {
  8097        let $pos = doc.resolve(pos);
* 8098        for (let d = $pos.depth;; d--) {
  8099            let before, after, index = $pos.index(d);
  8100            if (d == $pos.depth) {
  8101                before = $pos.nodeBefore;
  8102                after = $pos.nodeAfter;
  8103            }
  8104            else if (dir > 0) {
  8105                before = $pos.node(d + 1);
  8106                index++;
  8107                after = $pos.node(d).maybeChild(index);
  8108            }
```

```
8109                else {
8110                    before = $pos.node(d).maybeChild(index - 1);
8111                    after = $pos.node(d + 1);
8112                }
8113                if (before && !before.isTextblock && dist_joinable(before, after) &&
8114                    $pos.node(d).canReplace(index, index + 1))
8115                    return pos;
8116                if (d == 0)
8117                    break;
8118                pos = dir < 0 ? $pos.before(d) : $pos.after(d);
8119            }
8120        }
8121        function join(tr, pos, depth) {
8122            let step = new ReplaceStep(pos - depth, pos + depth, Slice.empty, true);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
53	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		8519:4 - 8525:4	

```
8516            // target depth, starting with the preferred depths.
8517            let preferredTargetIndex = targetDepths.indexOf(preferredTarget);
8518            let leftNodes = [], preferredDepth = slice.openStart;
* 8519          for (let content = slice.content, i = 0;; i++) {
8520                let node = content.firstChild;
8521                leftNodes.push(node);
8522                if (i == slice.openStart)
8523                    break;
8524                content = node.content;
8525            }
8526            // Back up preferredDepth to cover defining textblocks directly
8527            // above it, possibly skipping a non-defining textblock.
8528            for (let d = preferredDepth - 1; d >= 0; d--) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
54	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		9795:9 - 9795:17	

```
  9792          }
   9793     }
   9794
* 9795     function bindProps(obj, self, target) {
   9796          for (let prop in obj) {
   9797               let val = obj[prop];
   9798               if (val instanceof Function)
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
55	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		9892:4 - 9896:4	

```
  9889     const webkit_version = webkit ? +(/₩bAppleWebKit₩/(₩d+)/.exec(navigator.userAgent) || [0, 0])[1] : 0;
   9890
  9891     const domIndex = function (node) {
 * 9892          for (var index = 0;; index++) {
   9893               node = node.previousSibling;
   9894               if (!node)
   9895                    return index;
   9896          }
   9897     };
   9898     const parentNode = function (node) {
   9899          let parent = node.assignedSlot || node.parentNode;
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름

56	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		9921:4 - 9941:4	

```
 9918    };
  9919    const atomElements = /^(img|br|input|textarea|hr)$/i;
  9920    function scanFor(node, off, targetNode, targetOff, dir) {
* 9921        for (;;) {
  9922            if (node == targetNode && off == targetOff)
  9923                return true;
  9924            if (off == (dir < 0 ? 0 : nodeSize(node))) {
  9925                let parent = node.parentNode;
  9926                if (!parent || parent.nodeType != 1 || hasBlockDesc(node) ||
atomElements.test(node.nodeName) ||
  9927                    node.contentEditable == "false")
  9928                    return false;
  9929                off = domIndex(node) + (dir < 0 ? 0 : 1);
  9930                node = parent;
  9931            }
  9932            else if (node.nodeType == 1) {
  9933                node = node.childNodes[off + (dir < 0 ? -1 : 0)];
  9934                if (node.contentEditable == "false")
  9935                    return false;
  9936                off = dir < 0 ? nodeSize(node) : 0;
  9937            }
  9938            else {
  9939                return false;
  9940            }
  9941        }
  9942    }
  9943    function nodeSize(node) {
  9944        return node.nodeType == 3 ? node.nodeValue.length : node.childNodes.length;
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
57	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	

파일	/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js
발견 위치	10000:4 - 10033:4

```
    9997    function scrollRectIntoView(view, rect, startDOM) {
     9998        let scrollThreshold = view.someProp("scrollThreshold") || 0, scrollMargin =
view.someProp("scrollMargin") || 5;
     9999        let doc = view.dom.ownerDocument;
 * 10000        for (let parent = startDOM || view.dom;; parent = parentNode(parent)) {
    10001            if (!parent)
    10002                break;
    10003            if (parent.nodeType != 1)
    10004                continue;
    10005            let elt = parent;
    10006            let atTop = elt == doc.body;
    10007            let bounding = atTop ? windowRect(doc) : clientRect(elt);
    10008            let moveX = 0, moveY = 0;
    10009            if (rect.top < bounding.top + getSide(scrollThreshold, "top"))
    10010                moveY = -(bounding.top - rect.top + getSide(scrollMargin, "top"));
    10011            else if (rect.bottom > bounding.bottom - getSide(scrollThreshold, "bottom"))
    10012                moveY = rect.bottom - bounding.bottom + getSide(scrollMargin, "bottom");
    10013            if (rect.left < bounding.left + getSide(scrollThreshold, "left"))
    10014                moveX = -(bounding.left - rect.left + getSide(scrollMargin, "left"));
    10015            else if (rect.right > bounding.right - getSide(scrollThreshold, "right"))
    10016                moveX = rect.right - bounding.right + getSide(scrollMargin, "right");
    10017            if (moveX || moveY) {
    10018                if (atTop) {
    10019                    doc.defaultView.scrollBy(moveX, moveY);
    10020                }
    10021                else {
    10022                    let startX = elt.scrollLeft, startY = elt.scrollTop;
    10023                    if (moveY)
    10024                        elt.scrollTop += moveY;
    10025                    if (moveX)
    10026                        elt.scrollLeft += moveX;
    10027                    let dX = elt.scrollLeft - startX, dY = elt.scrollTop - startY;
    10028                    rect = { left: rect.left - dX, top: rect.top - dY, right: rect.right - dX,
bottom: rect.bottom - dY };
    10029                }
    10030            }
    10031            if (atTop)
```

```
10032                   break;
10033           }
10034   }
10035   // Store the scroll position of the editor's parent nodes, along with
10036   // the top position of an element near the top of the editor, which
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
58	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		10180:4 - 10196:4	

```
10177           // fall outside of. If so, we take the position before/after that
10178           // block. If not, we call `posFromDOM` on the raw node/offset.
10179           let outside = -1;
* 10180         for (let cur = node;;) {
10181               if (cur == view.dom)
10182                   break;
10183               let desc = view.docView.nearestDesc(cur, true);
10184               if (!desc)
10185                   return null;
10186               if (desc.node.isBlock && desc.parent) {
10187                   let rect = desc.dom.getBoundingClientRect();
10188                   if (rect.left > coords.left || rect.top > coords.top)
10189                       outside = desc.posBefore;
10190                   else if (rect.right < coords.left || rect.bottom < coords.top)
10191                       outside = desc.posAfter;
10192                   else
10193                       break;
10194               }
10195               cur = desc.dom.parentNode;
10196           }
10197           return outside > -1 ? outside : view.docView.posFromDOM(node, offset, 1);
10198   }
10199   function elementFromPoint(element, coords, box) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
59	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		10202:8 - 10214:8	

```
10199    function elementFromPoint(element, coords, box) {
10200        let len = element.childNodes.length;
10201        if (len && box.top < box.bottom) {
* 10202            for (let startI = Math.max(0, Math.min(len - 1, Math.floor(len * (coords.top - box.top) / (box.bottom - box.top)) - 2)), i = startI;;) {
10203                let child = element.childNodes[i];
10204                if (child.nodeType == 1) {
10205                    let rects = child.getClientRects();
10206                    for (let j = 0; j < rects.length; j++) {
10207                        let rect = rects[j];
10208                        if (inRect(coords, rect))
10209                            return elementFromPoint(child, coords, rect);
10210                    }
10211                }
10212                if ((i = (i + 1) % len) == startI)
10213                    break;
10214            }
10215        }
10216        return element;
10217    }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
60	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		11705:8 - 11726:8	

```
11702        let fI = frag.childCount, matched = new Map, matches = [];
11703        outer: while (fI > 0) {
11704            let desc;
```

```
* 11705                for (;;) {
  11706                     if (descI) {
  11707                          let next = curDesc.children[descI - 1];
  11708                          if (next instanceof MarkViewDesc) {
  11709                               curDesc = next;
  11710                               descI = next.children.length;
  11711                          }
  11712                          else {
  11713                               desc = next;
  11714                               descI--;
  11715                               break;
  11716                          }
  11717                     }
  11718                     else if (curDesc == parentDesc) {
  11719                          break outer;
  11720                     }
  11721                     else {
  11722                          // FIXME
  11723                          descI = curDesc.parent.children.indexOf(curDesc);
  11724                          curDesc = curDesc.parent;
  11725                     }
  11726                }
  11727                let node = desc.node;
  11728                if (!node)
  11729                     continue;
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
61	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		11757:4 - 11807:4	

```
  11754                return;
  11755           }
  11756           let decoIndex = 0, active = [], restNode = null;
* 11757           for (let parentIndex = 0;;) {
  11758                if (decoIndex < locals.length && locals[decoIndex].to == offset) {
```

```
11759                   let widget = locals[decoIndex++], widgets;
11760                   while (decoIndex < locals.length && locals[decoIndex].to == offset)
11761                       (widgets || (widgets = [widget])).push(locals[decoIndex++]);
11762                   if (widgets) {
11763                       widgets.sort(compareSide);
11764                       for (let i = 0; i < widgets.length; i++)
11765                           onWidget(widgets[i], parentIndex, !!restNode);
11766                   }
11767                   else {
11768                       onWidget(widget, parentIndex, !!restNode);
11769                   }
11770               }
11771               let child, index;
11772               if (restNode) {
11773                   index = -1;
11774                   child = restNode;
11775                   restNode = null;
11776               }
11777               else if (parentIndex < parent.childCount) {
11778                   index = parentIndex;
11779                   child = parent.child(parentIndex++);
11780               }
11781               else {
11782                   break;
11783               }
11784               for (let i = 0; i < active.length; i++)
11785                   if (active[i].to <= offset)
11786                       active.splice(i--, 1);
11787               while (decoIndex < locals.length && locals[decoIndex].from <= offset &&
locals[decoIndex].to > offset)
11788                   active.push(locals[decoIndex++]);
11789               let end = offset + child.nodeSize;
11790               if (child.isText) {
11791                   let cutAt = end;
11792                   if (decoIndex < locals.length && locals[decoIndex].from < cutAt)
11793                       cutAt = locals[decoIndex].from;
11794                   for (let i = 0; i < active.length; i++)
11795                       if (active[i].to < cutAt)
11796                           cutAt = active[i].to;
11797                   if (cutAt < end) {
```

```
11798                         restNode = child.cut(cutAt - offset);
11799                         child = child.cut(0, cutAt - offset);
11800                         end = cutAt;
11801                         index = -1;
11802                     }
11803                 }
11804                 let outerDeco = child.isInline && !child.isLeaf ? active.filter(d => !d.inline) :
active.slice();
11805                 onNode(child, outerDeco, deco.forChild(offset, child), index);
11806                 offset = end;
11807             }
11808     }
11809     // List markers in Mobile Safari will mysteriously disappear
11810     // sometimes. This works around that.
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
62	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		11820:4 - 11836:4	

```
11817         }
11818     }
11819     function nearbyTextNode(node, offset) {
 * 11820         for (;;) {
11821             if (node.nodeType == 3)
11822                 return node;
11823             if (node.nodeType == 1 && offset > 0) {
11824                 if (node.childNodes.length > offset && node.childNodes[offset].nodeType
== 3)
11825                     return node.childNodes[offset];
11826                 node = node.childNodes[offset - 1];
11827                 offset = nodeSize(node);
11828             }
11829             else if (node.nodeType == 1 && offset < node.childNodes.length) {
11830                 node = node.childNodes[offset];
11831                 offset = 0;
```

```
11832                 }
11833                 else {
11834                       return null;
11835                 }
11836           }
11837     }
11838     // Find a piece of text in an inline fragment, overlapping from-to
11839     function findTextInFragment(frag, text, from, to) {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
63	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		12167:4 - 12207:4	

```
12164           // node if possible. Issue prosemirror/prosemirror#832.
12165           if (gecko && node.nodeType == 1 && offset < nodeLen(node) &&
isIgnorable(node.childNodes[offset]))
12166                 force = true;
* 12167           for (;;) {
12168                 if (offset > 0) {
12169                       if (node.nodeType != 1) {
12170                             break;
12171                       }
12172                       else {
12173                             let before = node.childNodes[offset - 1];
12174                             if (isIgnorable(before)) {
12175                                   moveNode = node;
12176                                   moveOffset = --offset;
12177                             }
12178                             else if (before.nodeType == 3) {
12179                                   node = before;
12180                                   offset = node.nodeValue.length;
12181                             }
12182                             else
12183                                   break;
12184                       }
```

```
12185                 }
12186                 else if (isBlockNode(node)) {
12187                       break;
12188                 }
12189                 else {
12190                       let prev = node.previousSibling;
12191                       while (prev && isIgnorable(prev)) {
12192                             moveNode = node.parentNode;
12193                             moveOffset = domIndex(prev);
12194                             prev = prev.previousSibling;
12195                       }
12196                       if (!prev) {
12197                             node = node.parentNode;
12198                             if (node == view.dom)
12199                                   break;
12200                             offset = 0;
12201                       }
12202                       else {
12203                             node = prev;
12204                             offset = nodeLen(node);
12205                       }
12206                 }
12207           }
12208           if (force)
12209                 setSelFocus(view, sel, node, offset);
12210           else if (moveNode)
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
64	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		12222:4 - 12256:4	

```
12219                 return;
12220           let len = nodeLen(node);
12221           let moveNode, moveOffset;
* 12222         for (;;) {
```

```
12223                if (offset < len) {
12224                    if (node.nodeType != 1)
12225                        break;
12226                    let after = node.childNodes[offset];
12227                    if (isIgnorable(after)) {
12228                        moveNode = node;
12229                        moveOffset = ++offset;
12230                    }
12231                    else
12232                        break;
12233                }
12234                else if (isBlockNode(node)) {
12235                    break;
12236                }
12237                else {
12238                    let next = node.nextSibling;
12239                    while (next && isIgnorable(next)) {
12240                        moveNode = next.parentNode;
12241                        moveOffset = domIndex(next) + 1;
12242                        next = next.nextSibling;
12243                    }
12244                    if (!next) {
12245                        node = node.parentNode;
12246                        if (node == view.dom)
12247                            break;
12248                        offset = len = 0;
12249                    }
12250                    else {
12251                        node = next;
12252                        offset = 0;
12253                        len = nodeLen(node);
12254                    }
12255                }
12256            }
12257            if (moveNode)
12258                setSelFocus(view, sel, moveNode, moveOffset);
12259        }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
65	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		13908:13 - 13908:18	

```
13905    }
13906    function mapAndGatherRemainingDecorations(children, oldChildren, decorations,
mapping, offset, oldOffset, options) {
13907        // Gather all decorations from the remaining marked children
* 13908        function gather(set, oldOffset) {
13909            for (let i = 0; i < set.local.length; i++) {
13910                let mapped = set.local[i].map(mapping, offset, oldOffset);
13911                if (mapped)
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
66	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/toastui-editor-all.js	
발견 위치		15830:8 - 15835:8	

```
15827        }
15828        if (canDelAfter && textblockAt(after, "start", true) && textblockAt(before, "end")) {
15829            let at = before, wrap = [];
* 15830            for (;;) {
15831                wrap.push(at);
15832                if (at.isTextblock)
15833                    break;
15834                at = at.lastChild;
15835            }
15836            let afterText = after, afterDepth = 1;
15837            for (; !afterText.isTextblock; afterText = afterText.firstChild)
15838                afterDepth++;
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
67	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/libs/codemirror.js	
발견 위치		1970:6 - 1984:6	

```
 1967            if (pos >= len) { break }
  1968
  1969            var upto = Math.min(len, nextChange);
* 1970             while (true) {
  1971              if (text) {
  1972                var end = pos + text.length;
  1973                if (!collapsed) {
  1974                  var tokenText = end > upto ? text.slice(0, upto - pos) : text;
  1975                  builder.addToken(builder, tokenText, style ? style + spanStyle : spanStyle,
  1976                                   spanStartStyle, pos + tokenText.length == nextChange ?
spanEndStyle : "", css, attributes);
  1977                }
  1978                if (end >= upto) {text = text.slice(upto - pos); pos = upto; break}
  1979                pos = end;
  1980                spanStartStyle = "";
  1981              }
  1982              text = allText.slice(at, at = styles[i++]);
  1983              style = interpretTokenStyle(styles[i++], builder.cm.options);
  1984            }
  1985          }
  1986        }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
68	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/libs/codemirror.js	
발견 위치		5309:4 - 5322:4	

```
 5306          if (i == source.length) { return }
  5307          hist.lastOrigin = hist.lastSelOrigin = null;
  5308
* 5309          for (;;) {
  5310            event = source.pop();
```

```
5311            if (event.ranges) {
5312              pushSelectionToHistory(event, dest);
5313              if (allowSelectionOnly && !event.equals(doc.sel)) {
5314                setSelection(doc, event, {clearRedo: false});
5315                return
5316              }
5317              selAfter = event;
5318            } else if (suppress) {
5319              source.push(event);
5320              return
5321            } else { break }
5322          }
5323
5324        // Build up a reverse change object to add to the opposite history
5325         // stack (redo when undoing, and vice versa).
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
69	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/libs/codemirror.js	
발견 위치		8311:15 - 8316:8	

```
8308            var before = 0, after = (styles.length - 1) / 2, ch = pos.ch;
8309              var type;
8310              if (ch == 0) { type = styles[2]; }
* 8311            else { for (;;) {
8312                var mid = (before + after) >> 1;
8313                if ((mid ? styles[mid * 2 - 1] : 0) >= ch) { after = mid; }
8314                else if (styles[mid * 2 + 1] < ch) { before = mid + 1; }
8315                else { type = styles[mid * 2 + 2]; break }
8316              } }
8317              var cut = type ? type.indexOf("overlay ") : -1;
8318              return cut < 0 ? type : cut == 0 ? null : type.slice(0, cut - 1)
8319            },
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름

70	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/libs/toastui-editor-viewer.js	
발견 위치		4993:9 - 4993:15	

```
 4990     */
  4991
 4992
* 4993    function tracker(runner, renderer) {
  4994        var subContent = '';
  4995        var node = runner.getNode();
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
71	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/libs/toastui-editor.js	
발견 위치		5112:9 - 5112:15	

```
 5109     */
  5110
 5111
* 5112    function tracker(runner, renderer) {
  5113        var subContent = '';
  5114        var node = runner.getNode();
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
72	0021_NONTERM	4	종료되지 않는 반복문 또는 재귀 함수
CWE/CERT		CWE-674	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/libs/toastui-editor.js	
발견 위치		8766:9 - 8766:17	

```
 8763     */
  8764
 8765
* 8766    function fixNumber(lineNumber, prevIndentLength, startIndex, cm) {
  8767        var indent, delimiter, text, indentLength;
```

```
8768        var index = startIndex;
8769        var lineText = cm.getLine(lineNumber);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

4.7. 오류 상황 대응 부재 [0023_NOACTION]

규칙 코드	0023_NOACTION
CWE/CERT	CWE-390,ERR00-J
관련 규칙	행안부 SW 보안약점 2021 - 오류상황 대응 부재
규칙 설명	오류가 발생할 수 있는 부분을 확인하였으나, 이러한 오류에 대하여 예외 처리를 하지 않을 경우, 공격자는 오류 상황을 악용하여 개발자가 의도하지 않은 방향으로 프로그램이 동작하도록 할 수 있습니다.

올바른 예제

```
1   protected Element createContent(WebSession s) {
 2        ......
 3        try {
 4             username = s.getParser().getRawParameter(USERNAME);
 5             password = s.getParser().getRawParameter(PASSWORD);
 6             if (!"webgoat".equals(username) || !password.equals("webgoat")) {
 7                  s.setMessage("Invalid username and password entered.");
 8                  return (makeLogin(s));
 9             }
10   **       } catch (NullPointerException e) {
11             // 예외 사항에 대해 적절한 조치를 수행하여야 합니다.
12   **          s.setMessage(e.getMessage());
13   **          return (makeLogin(s));
14        }
```

예외를 포착(catch)한 후, 각각의 예외 사항(Exception)에 대하여 적절하게 처리해야 합니다.

잘못된 예제

```
1   protected Element createContent(WebSession s) {
 2        ......
 3        try {
 4             username = s.getParser().getRawParameter(USERNAME);
 5             password = s.getParser().getRawParameter(PASSWORD);
```

```
 6              if (!"webgoat".equals(username) || !password.equals("webgoat")) {
 7                   s.setMessage("Invalid username and password entered.");
 8                   return (makeLogin(s));
 9              }
10   **     } catch (NullPointerException e) {
11         // 요청 파라미터에 PASSWORD 가 존재하지 않을 경우 Null Pointer Exception 이 발생하고 해당 오류에 대한 대응이 존재하지 않아 인증이 된 것으로 처리
12   }
```

try 블록에서 발생하는 오류를 포착(catch)하고 있지만, 그 오류에 대해서 아무 조치를 하고 있지 않음을 보여줍니다. 아무 조치가 없으므로 프로그램이 계속 실행되기 때문에 프로그램에서 어떤 일이 일어났는지 전혀 알 수 없게 됩니다.

순번	규칙 코드	위험도	규칙 이름
73	0023_NOACTION	4	오류 상황 대응 부재
CWE/CERT		CWE-390,ERR00-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/StoneUtil.java	
발견 위치		275:29 - 276:3	

```
 272              try {
  273                   Date date = dateformat.parse(value);
  274                   result = new SimpleDateFormat("yyyy-MM-dd").format(date);
* 275              } catch (ParseException e) {
  276              }
  277              return result;
  278
 279         }
```

코드 수정 제안	275행 Catch 문에 아무 조치를 하지 않기 때문에 프로그램에서 어떤 일이 일어났는지 전혀 알 수 없게 됩니다. 275행 Catch 문에 대한 적절한 처리가 필요 합니다.

순번	규칙 코드	위험도	규칙 이름
74	0023_NOACTION	4	오류 상황 대응 부재
CWE/CERT		CWE-390,ERR00-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/security/CsrfTokenAdder.java	
발견 위치		188:26 - 189:3	

```
185              capture = new ByteArrayOutputStream(response.getBufferSize());
```

```
  186                 try {
  187                       capture.close();
* 188                 } catch (IOException e) {
  189                 } finally {
  190                       capture.close();
  191                 }
  192
```

코드 수정 제안	188행 Catch 문에 아무 조치를 하지 않기 때문에 프로그램에서 어떤 일이 일어났는지 전혀 알 수 없게 됩니다. 188행 Catch 문에 대한 적절한 처리가 필요 합니다.

순번	규칙 코드	위험도	규칙 이름
75	0023_NOACTION	4	오류 상황 대응 부재
CWE/CERT		CWE-390,ERR00-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/security/CsrfTokenAdder.java	
발견 위치		243:29 - 244:4	

```
  240                 if (writer == null) {
  241                       try {
  242                             writer = new PrintWriter(new OutputStreamWriter(capture,
getCharacterEncoding()));
* 243                       } catch (IOException ioe) {
  244                       } finally {
  245                             writer.close();
  246                       }
  247
```

코드 수정 제안	243행 Catch 문에 아무 조치를 하지 않기 때문에 프로그램에서 어떤 일이 일어났는지 전혀 알 수 없게 됩니다. 243행 Catch 문에 대한 적절한 처리가 필요 합니다.

순번	규칙 코드	위험도	규칙 이름
76	0023_NOACTION	4	오류 상황 대응 부재
CWE/CERT		CWE-390,ERR00-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/utils/DateValidator.java	

발견 위치	39:39 - 41:5

```
 36                      .from(localDateTime.atZone(ZoneId.systemDefault())
 37                              .toInstant());
 38
* 39          } catch (DateTimeParseException e) {
 40
 41          }
 42            return null;
 43
 44        }
```

코드 수정 제안	39행 Catch 문에 아무 조치를 하지 않기 때문에 프로그램에서 어떤 일이 일어났는지 전혀 알 수 없게 됩니다. 39행 Catch 문에 대한 적절한 처리가 필요 합니다.

4.8. 부적절한 예외 처리 [0024_NOCHKERR]

규칙 코드	0024_NOCHKERR
CWE/CERT	CWE-754,ERR06-J
관련 규칙	행안부 SW 보안약점 2021 - 부적절한 예외 처리
규칙 설명	프로그램 수행 중에 함수의 결괏값에 대한 적절한 처리 또는 예외 상황에 대한 조건을 적절하게 검사하지 않을 경우, 예기치 않은 문제를 야기할 수 있습니다.

올바른 예제

```
1   try {
 2          ...
 3          reader = new BufferedReader(new InputStreamReader(url.openStream()));
 4          String line = reader.readLine();
 5          SimpleDateFormat format = new SimpleDateFormat("MM/DD/YY");
 6          Date date = format.parse(line);
 7          // 발생할 수 있는 오류의 종류와 순서에 맞춰서 예외처리 합니다.
 8   **} catch (MalformedURLException e) {
 9          System.err.println("MalformedURLException : " + e.getMessage());
10   **} catch (IOException e) {
11          System.err.println("IOException : " + e.getMessage());
12   **} catch (ParseException e) {
13          System.err.println("ParseException : " + e.getMessage());
```

```
14  }
```

발생 가능한 예외를 세분화하고 발생 가능한 순서에 따라 예외를 처리하고 있습니다.

잘못된 예제

```
1   try {
 2          ...
 3          reader = new BufferedReader(new InputStreamReader(url.openStream()));
 4          String line = reader.readLine();
 5          SimpleDateFormat format = new SimpleDateFormat("MM/DD/YY");
 6          Date date = format.parse(line);
 7          // 예외처리를 세분화 할 수 있음에도 광범위하게 사용하여 예기치 않은 문제가 발생 할 수 있습니다.
 8   **} catch (Exception e) {
 9          System.err.println("Exception : " + e.getMessage());
10   }
```

try 블록에서 다양한 예외가 발생할 수 있음에도 불구하고 예외를 세분화하지 않고 광범위한 예외 클래스인 Exception 을 사용하여 예외를 처리하고 있습니다.

순번	규칙 코드	위험도	규칙 이름
77	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/DocentBotController.java	
발견 위치		60:17 - 60:28	

```
 57                 paramsMap.putWithNotEmpty("keyword", params.getKeyword());
  58
 59                 list = docentService.list(paramsMap, paging.getPaging());
* 60             } catch (Exception e) {
  61                 log.error(e.getMessage(), e);
  62             }
```

코드 수정 제안	60행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 60행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
78	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/DocentBotController.java	
발견 위치		192:17 - 192:28	

```
189                 }
 190
 191                 docentService.save(form, authenticationFacade.getLoginUserSeq());
* 192             } catch (Exception e) {
 193                 log.error(e.getMessage(), e);
 194
 195                 model.addAttribute("menuCode", menuCode);
```

코드 수정 제안	192행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 192행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
79	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/ExhibitController.java	
발견 위치		129:13 - 129:24	

```
126             rttr.addFlashAttribute("result_code", GlobalConstant.CRUD_TYPE.SAVE);
 127
 128             return "redirect:/exhibit?" + exhibit.getPageParams();
* 129         } catch (Exception e) {
 130             log.error(e.getMessage());
 131             return "error/error";
 132         }
```

코드 수정 제안	129행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 129행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름

80	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/ExhibitController.java	
발견 위치		143:13 - 143:24	

```
 140            rttr.addFlashAttribute("result_code", GlobalConstant.CRUD_TYPE.DELETE);
 141
 142            return "redirect:/exhibit?" + pageParams;
* 143          } catch (Exception e) {
 144              log.error(e.getMessage());
 145              return "error/error";
 146          }
```

코드 수정 제안	143행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 143행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
81	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/NewsroomController.java	
발견 위치		115:13 - 115:24	

```
 112            rttr.addFlashAttribute("result_code", GlobalConstant.CRUD_TYPE.SAVE);
 113
 114            return "redirect:/newsroom?" + newsroom.getPageParams();
* 115          } catch (Exception e) {
 116              log.error(e.getMessage());
 117              return "error/error";
 118          }
```

코드 수정 제안	115행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 115행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
82	0024_NOCHKERR	4	부적절한 예외 처리

CWE/CERT	CWE-754,ERR06-J
파일	/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/NewsroomController.java
발견 위치	129:13 - 129:24

```
 126            rttr.addFlashAttribute("result_code", GlobalConstant.CRUD_TYPE.DELETE);
  127
 128            return "redirect:/newsroom?" + pageParams;
* 129          } catch (Exception e) {
  130            log.error(e.getMessage());
  131            return "error/error";
  132          }
```

코드 수정 제안	129행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 129행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
83	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		151:11 - 151:22	

```
 148                    compressZipDirectory(origin, zipOutputSteam, file, topFile.getName() +
File.separator);
  149                }
  150              }
* 151            } catch (Exception e) {
  152              log.error("", e);
  153            } finally {
  154              try {
```

코드 수정 제안	151행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 151행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
84	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	

파일	/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java
발견 위치	326:11 - 326:22

```
 323                         encodedFilename = URLEncoder.encode(encodedFilename,
StandardCharsets.UTF_8.name()).replaceAll("₩₩+",
 324                                         "%20");
 325                     }
 * 326                 } catch (Exception e) {
 327                     log.error("", e);
 328                     encodedFilename = filename;
 329                 }
```

코드 수정 제안	326행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 326행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
85	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		475:12 - 475:23	

```
 472                         encodeFilename = URLEncoder.encode(encodeFilename,
StandardCharsets.UTF_8.name()).replaceAll("₩₩+",
 473                                         "%20");
 474                     }
 * 475                 } catch (Exception e) {
 476                     log.error("", e);
 477                     encodeFilename = filename;
 478                 }
```

코드 수정 제안	475행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 475행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
86	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	

파일	/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/StoneUtil.java
발견 위치	228:10 - 228:21

```
 225              try{
  226                   SimpleDateFormat sDateFormat = new
SimpleDateFormat(GlobalConstant.EGOV_DATE_PATTERN);
  227                   return new DateTime(sDateFormat.parse(strDate));
* 228              }catch (Exception e){
  229                   return null;
  230              }
```

코드 수정 제안	228행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 228행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
87	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/StringUtil.java	
발견 위치		74:12 - 74:23	

```
 71              if (obj != null) {
  72                   try {
  73                        result = Integer.parseInt(obj.toString());
* 74                   } catch (Exception e) {
  75                        log.error("getInteger error", e);
  76                        result = 0;
  77                   }
```

코드 수정 제안	74행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 74행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
88	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/StringUtil.java	
발견 위치		87:12 - 87:23	

```
 84              if (obj != null) {
 85                  try {
 86                      result = Double.parseDouble(obj.toString());
* 87                  } catch (Exception e) {
 88                      log.error("getInteger error", e);
 89                      result = 0;
 90                  }
```

코드 수정 제안	87행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 87행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
89	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/StringUtil.java	
발견 위치		100:12 - 100:23	

```
 97              if (obj != null) {
 98                  try {
 99                      result = "true".equals(obj.toString().toLowerCase()) ||
Integer.parseInt(obj.toString()) == 1;
* 100                 } catch (Exception e) {
 101                     result = false;
 102                 }
 103             }
```

코드 수정 제안	100행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 100행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
90	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/ValidatorUtil.java	
발견 위치		39:11 - 39:22	

```
36          }
```

```
  37                      }
  38                  }
* 39              } catch (Exception e) {
  40                  if (log.isErrorEnabled()) {
  41                      log.error("validate : " + e.getLocalizedMessage());
  42                  }
```

코드 수정 제안	39행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 39행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
91	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/ValidatorUtil.java	
발견 위치		78:11 - 78:22	

```
 75                      methods.add(method.getName());
  76                  }
  77              }
* 78          } catch (Throwable e) {
  79              if (log.isErrorEnabled()) {
  80                  log.error("getCollectionTypeMethodNames : " + e.getLocalizedMessage());
  81              }
```

코드 수정 제안	78행 java.lang.Throwable 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 78행 java.lang.Throwable 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
92	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelExporter.java	
발견 위치		107:19 - 107:30	

```
 104              } else {
 105                  val = TypeConverterManager.convertType(BeanUtil.getProperty(objects[i],
```

```
properties[j]), String.class);
    106                     }
 * 107                  } catch (Exception e) {
    108                     val = "";
    109                  }
```

코드 수정 제안	107행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 107행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
93	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		81:11 - 81:22	

```
   78                  //headerPrint(parser);
   79                    rowsPrint(parser);
   80                    outputExcel(parser, response, request);
 * 81                } catch (Throwable e) {
   82                    e.printStackTrace();
   83                }
   84            }
```

코드 수정 제안	81행 java.lang.Throwable 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 81행 java.lang.Throwable 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
94	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		217:16 - 217:27	

```
  214                                 try {
  215                                     formatString =
StringUtil.getAccountFormat(cells.getValue("ExcelSymbol"),
```

```
  216                                                 Integer.parseInt(cells.getValue("Precisions")));
* 217                                         } catch (Exception e) {
  218                                               formatString = "#,##0";
  219                                          }
  220                                     }
```

코드 수정 제안	217행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 217행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
95	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		235:13 - 235:24	

```
  232                                     }
  233                                 }
  234                             }
* 235                         } catch (Exception e) {
  236                               cell.setCellValue(cells.getValue(cols[0][j].getData()));
  237                          }
```

코드 수정 제안	235행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 235행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
96	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		289:11 - 289:22	

```
  286                 os = response.getOutputStream();
  287                   bos = new BufferedOutputStream(os);
  288                   workbook.write(bos);
* 289               } catch (Exception e) {
```

```
290                     log.error("", e);
291                 } finally {
292                     if (bos != null) {
```

코드 수정 제안	289행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 289행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
97	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AuthenticationController.java	
발견 위치		191:17 - 191:28	

```
188                     logoutResponse.setStatus(200);
189
190                     return ResponseEntity.ok(logoutResponse);
* 191                 } catch (Exception e) {
192                     log.error(e.getMessage(), e);
193
194                     return internalServerError();
```

코드 수정 제안	191행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 191행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
98	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AuthenticationController.java	
발견 위치		219:17 - 219:28	

```
216                     } else {
217                         return badRequestIdUsed();
218                     }
* 219                 } catch (Exception e) {
220                     log.error(e.getMessage(), e);
```

```
 221
 222                    return internalServerError();
```

코드 수정 제안	219행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 219행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
99	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AuthenticationController.java	
발견 위치		267:17 - 267:28	

```
 264
 265                        return ResponseEntity.ok(CommonResponse.ok());
  266                    }
* 267                } catch (Exception e) {
  268                    log.error(e.getMessage(), e);
  269
 270                    return internalServerError();
```

코드 수정 제안	267행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 267행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
100	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AuthenticationController.java	
발견 위치		336:17 - 336:28	

```
 333                    } else {
  334                        return badRequestIdUsed();
  335                    }
* 336                } catch (Exception e) {
  337                    log.error(e.getMessage(), e);
  338
```

```
339                     return internalServerError();
```

코드 수정 제안	336행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 336행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
101	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AvatarController.java	
발견 위치		64:17 - 64:28	

```
 61                     return ok();
  62
 63
* 64                 } catch (Exception e) {
  65                     log.error(e.getMessage(), e);
  66
 67                     return internalServerError();
```

코드 수정 제안	64행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 64행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
102	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AvatarController.java	
발견 위치		102:17 - 102:28	

```
  99                     return ok();
  100
 101
* 102                 } catch (Exception e) {
  103                     log.error(e.getMessage(), e);
  104                     return internalServerError();
  105                 }
```

코드 수정 제안	102행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 102행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
103	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/CommonController.java	
발견 위치		40:13 - 40:24	

```
 37            } else {
 38                throw new NotFoundException();
 39            }
* 40        } catch (Exception e) {
 41            return null;
 42        }
 43    }
```

코드 수정 제안	40행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 40행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
104	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/CommonController.java	
발견 위치		83:13 - 83:24	

```
 80                len = fileIn.read(buf, 0, 1024);
 81                out.flush();
 82            }
* 83        } catch (Exception e) {
 84            log.error(e.getMessage(), e);
 85            throw new IOException(e);
 86        } finally {
```

코드 수정 제안	83행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 83행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
105	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		73:13 - 73:24	

```
 70            } else {
 71                return requiredFieldBadRequest(errorField);
 72            }
* 73          } catch (Exception e) {
 74            log.error(e.getMessage(), e);
 75            return internalServerError();
 76          }
```

코드 수정 제안	73행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 73행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
106	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		114:13 - 114:24	

```
111            } else {
 112               return requiredFieldBadRequest(errorField);
 113            }
* 114          } catch (Exception e) {
 115            log.error(e.getMessage(), e);
 116            return internalServerError();
 117          }
```

코드 수정 제안	114행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은

	문제를 야기할 수 있습니다. 114행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
107	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		163:13 - 163:24	

```
  160             docentService.updateDocentStatus(docent);
   161             return ResponseEntity.ok(CommonResponse.ok());
   162
* 163         } catch (Exception e) {
   164             log.error(e.getMessage(), e);
   165             return internalServerError();
   166         }
```

코드 수정 제안	163행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 163행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
108	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		194:13 - 194:24	

```
  191             List<DocentVO> docentList = docentService.selectPagination(params,
paging.getPaging());
   192             PagingResponse pagingResponse = this.convertDocentPagingResponse(docentList,
paging, userId, false);
   193             return ResponseEntity.ok(pagingResponse);
 * 194         } catch (Exception e) {
   195             log.error(e.getMessage(), e);
   196             return internalServerError();
   197         }
```

코드 수정 제안	194행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 194행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
109	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		222:13 - 222:24	

```
 219            result.putAll(map);
  220
 221            return ok(result);
* 222          } catch (Exception e) {
  223            log.error(e.getMessage(), e);
  224            return internalServerError();
  225          }
```

코드 수정 제안	222행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 222행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
110	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		237:13 - 237:24	

```
  234            List<DocentVO> docentList = docentService.selectPagination(params,
paging.getPaging());
   235            PagingResponse pagingResponse = this.convertDocentPagingResponse(docentList,
paging, null, true);
   236            return ResponseEntity.ok(pagingResponse);
 * 237          } catch (Exception e) {
   238            log.error(e.getMessage(), e);
   239            return internalServerError();
   240          }
```

코드 수정 제안	237행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 237행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
111	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		256:13 - 256:24	

```
  253
  254             docentService.deleteByPK(Integer.parseInt(docentSeq));
   255             return ResponseEntity.ok(CommonResponse.ok());
* 256         } catch (Exception e) {
   257             log.error(e.getMessage(), e);
   258             return internalServerError();
   259         }
```

코드 수정 제안	256행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 256행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
112	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		279:13 - 279:24	

```
  276             List<DocentPersonVO> docentPersonList =
docentPersonService.selectPagination(params, paging.getPaging());
   277             PagingResponse pagingResponse =
this.convertDocentPersonPagingResponse(docentPersonList, paging, docentSeq);
   278             return ResponseEntity.ok(pagingResponse);
 * 279         } catch (Exception e) {
   280             log.error(e.getMessage(), e);
   281             return internalServerError();
```

```
282         }
```

코드 수정 제안	279행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 279행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
113	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		348:13 - 348:24	

```
345         try {
 346             DateTime.parse(date, DateTimeFormat.forPattern(format));
 347             return false;
* 348         } catch (Exception e) {
 349             return true;
 350         }
 351     }
```

코드 수정 제안	348행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 348행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
114	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/LogController.java	
발견 위치		92:13 - 92:24	

```
 89             } else {
 90                 return requiredFieldBadRequest(errorField);
 91             }
* 92         } catch (Exception e) {
 93             log.error(e.getMessage(), e);
 94             return internalServerError();
 95         }
```

코드 수정 제안	92행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 92행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
115	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/LogController.java	
발견 위치		122:13 - 122:24	

```
 119            response.setDashboard(dashboard);
 120
 121            return ResponseEntity.ok(response);
* 122          } catch (Exception e) {
 123            log.error(e.getMessage(), e);
 124            return internalServerError();
 125          }
```

코드 수정 제안	122행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 122행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
116	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/LogController.java	
발견 위치		157:13 - 157:24	

```
 154          try {
 155            DateTime.parse(date, DateTimeFormat.forPattern(format));
 156            return false;
* 157          } catch (Exception e) {
 158            return true;
 159          }
 160        }
```

코드 수정 제안	157행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 157행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
117	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/MessageController.java	
발견 위치		81:17 - 81:28	

```
 78                  }else{
  79                        return requiredFieldBadRequest(errorField);
  80                  }
* 81              } catch (Exception e) {
  82                  log.error(e.getMessage(), e);
  83                  return internalServerError();
  84              }
```

코드 수정 제안	81행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 81행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

순번	규칙 코드	위험도	규칙 이름
118	0024_NOCHKERR	4	부적절한 예외 처리
CWE/CERT		CWE-754,ERR06-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/utils/RequestUtils.java	
발견 위치		40:17 - 40:28	

```
 37              User user;
  38              try {
  39                  user = (User)
SecurityContextHolder.getContext().getAuthentication().getPrincipal();
* 40              } catch (Exception e) {
  41                  user = null;
  42              }
  43              return user;
```

코드 수정 제안	40행 java.lang.Exception 은 모든 예외를 포괄하여 예기치 않은 문제를 야기할 수 있습니다. 40행 java.lang.Exception 대신 구체적인 예외처리를 합니다.

4.9. 널(Null) 포인터 역참조 [0025_USENULL]

규칙 코드	0025_USENULL
CWE/CERT	CWE-476,EXP01-J
관련 규칙	행안부 SW 보안약점 2021 - 널(Null) 포인터 역참조
규칙 설명	널 포인터 역참조는 '일반적으로 그 객체가 널(Null)이 될 수 없다'라고 하는 가정을 위반했을 때 발생합니다. 공격자가 의도적으로 널 포인터 역참조를 발생시키는 경우, 그 결과 발생하는 예외 상황을 이용하여 추후의 공격을 계획하는 데 사용될 수 있습니다.

올바른 예제

```
1   public static int cardinality (Object obj, final Collection col) {
 2        int count = 0;
 3        if (col == null) {
 4             return count;
 5        }
 6        Iterator it = col.iterator();
 7
8        while (it.hasNext()) {
 9             Object elt = it.next();
10  **           if ((null == obj &amp;amp;&amp;amp; null == elt) || (null != obj &amp;amp;&amp;amp; obj.equals(elt))) {
11                count++;
12             }
13        }
14        return count;
15  }
```

obj 가 null 인지 검사 후 참조해야 합니다.

잘못된 예제

```
1   public static int cardinality (Object obj, final Collection col) {
 2        int count = 0;
```

```
 3        if (col == null) {
 4             return count;
 5        }
 6        Iterator it = col.iterator();
 7        while (it.hasNext()) {
 8             Object elt = it.next();
 9             // obj 가 null 이고 elt 가 null 이 아닐 경우, Null.equals 가 되어 널(Null) 포인터
역참조가 발생합니다.
10   **           if ((null == obj &amp;amp;&amp;amp; null == elt) || obj.equals(elt)) {
11                  count++;
12             }
13        }
14        return count;
15   }
```

obj 가 null 이고, elt 가 null 이 아닌 경우 널(Null) 포인터 역참조가 발생합니다.

순번	규칙 코드	위험도	규칙 이름
119	0025_USENULL	5	널(Null) 포인터 역참조
CWE/CERT		CWE-476,EXP01-J	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		214:20 - 214:48	

```
 211                        ZipEntry entry = new ZipEntry(file.getName());
  212                        zout.putNextEntry(entry);
  213                        int count;
* 214                        while ((count = origin.read(data, 0, BUFFER)) != -1) {
  215                              zout.write(data, 0, count);
  216                        }
  217                   } catch (IOException e) {
```

코드 수정 제안	214 행 read(..) 함수 호출 시 137 행 origin 변수 선언에서 null 이 배정된 채로 사용하게되면 세크멘테이션 폴트나 널 포인터 예외가 발생할 수 있습니다. 214 행 read(..) 함수 호출 전 null 검사를 하거나 예외처리를 하세요.

순번	규칙 코드	위험도	규칙 이름
120	0025_USENULL	5	널(Null) 포인터 역참조

CWE/CERT	CWE-476,EXP01-J
파일	/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java
발견 위치	334:17 - 334:58

```
  331                     ClientAnchor anchor = new HSSFClientAnchor(4, 2, 4, 2, (short) cell, row, (short) cell, row);
    332                     anchor.setAnchorType(ClientAnchor.DONT_MOVE_AND_RESIZE);
    333
* 334                     Picture pic = drawing.createPicture(anchor, pictureIdx);
    335     //                  double scale = 1.0;
    336                     pic.resize();
    337                 } finally {
```

코드 수정 제안	334 행 createPicture(..) 함수 호출 시 165 행 drawing 변수 선언에서 null 이 배정된 채로 사용하게되면 세크멘테이션 폴트나 널 포인터 예외가 발생할 수 있습니다. 334 행 createPicture(..) 함수 호출 전 null 검사를 하거나 예외처리를 하세요.

순번	규칙 코드	위험도	규칙 이름
121	0025_USENULL	5	널(Null) 포인터 역참조
CWE/CERT		CWE-476,EXP01-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/AvatarController.java	
발견 위치		58:29 - 58:69	

```
  55                  avatar.setItemCodes(form.getItemCodes());
    56                  avatar.setBodyCode(form.getBodyCode());
    57
* 58                  Integer useSeq = RequestUtils.getLoginUser().getUserSeq();
    59                  userAvatarService.addOrUpdate(avatar, form.getThumbFile(), useSeq);
    60
  61                  return ok();
```

코드 수정 제안	58 행 getUserSeq(..) 함수 호출 시 41 행 user 변수 사용에서 null 이 배정된 채로 사용하게되면 세크멘테이션 폴트나 널 포인터 예외가 발생할 수 있습니다. 58 행 getUserSeq(..) 함수 호출 전 null 검사를 하거나

	예외처리를 하세요.

순번	규칙 코드	위험도	규칙 이름
122	0025_USENULL	5	널(Null) 포인터 역참조
CWE/CERT		CWE-476,EXP01-J	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/DocentController.java	
발견 위치		48:29 - 48:69	

```
 45      @PostMapping(value = "/docent")
 46       public ResponseEntity<?> register(DocentForm form) {
 47          try {
* 48            Integer loginUserSeq = RequestUtils.getLoginUser().getUserSeq();
 49            if (form.getUser_id() != null) {
 50               form.setUser_id(form.getUser_id().trim());
 51            }
```

코드 수정 제안	48 행 getUserSeq(..) 함수 호출 시 41 행 user 변수 사용에서 null 이 배정된 채로 사용하게되면 세크멘테이션 폴트나 널 포인터 예외가 발생할 수 있습니다. 48 행 getUserSeq(..) 함수 호출 전 null 검사를 하거나 예외처리를 하세요.

4.10. 부적절한 자원 해제 [0026_RESLEAK]

규칙 코드	0026_RESLEAK
CWE/CERT	CWE-404,CWE-772
관련 규칙	행안부 SW 보안약점 2021 - 부적절한 자원 해제
규칙 설명	프로그램의 자원, 예를 들면 열린 파일디스크립터(Open File descriptor), 힙 메모리(Heap Memory), 소켓(Socket) 등은 유한한 자원입니다. 이러한 자원을 할당받아 사용한 후, 더 이상 사용하지 않는 경우에는 적절히 반환하여야 하는데, 프로그램 오류 또는 에러로 사용이 끝난 자원을 반환하지 못하는 경우입니다.

올바른 예제

```
1   InputStream in = null;
 2   OutputStream out = null;
 3   try {
```

```
 4        in = new FileInputStream(inputFile);
 5        out = new FileOutputStream(outputFile);
 6        ...
 7        FileCopyUtils.copy(fis, os);
 8    } catch (IOException e) {
 9        logger.error(e);
10        // 항상 수행되는 finally 블록에서 할당받은 모든 자원에 대해 각각 null 검사를 수행 후 예외처리를 하여 자원을 해제하여야 합니다.
11    **} finally {
12    **      if (in != null) {
13    **          try {
14    **              in.close();
15    **          } catch (IOException e) {
16    **              logger.error(e);
17    **          }
18    **      }
19    **      if (out != null) {
20    **          try {
21    **              out.close();
22    **          } catch (IOException e) {
23    **              logger.error(e);
24    **          }
25    **      }
26    **}
```

예외 상황이 발생하여 함수가 종료될 때, 예외의 발생 여부와 상관없이 할당받은 자원을 finally 블록에서 반환하여 사용한 자원은 반드시 반환하도록 합니다.

잘못된 예제

```
1    InputStream in = null;
 2    OutputStream out = null;
 3    try {
 4    **      in = new FileInputStream(inputFile);
 5    **      out = new FileOutputStream(outputFile);
 6        ...
 7        FileCopyUtils.copy(fis, os);
 8        // 자원반환 실행 전에 오류가 발생할 경우 자원이 반환되지 않으며, 할당된 모든 자원을 반환해야 합니다.
```

```
 9  **      in.close();
10  **      out.close();
11  } catch (IOException e) {
12          logger.error(e);
13  }
```

try 구문 내 처리 중 오류가 발생할 경우, close()메소드가 실행되지 않아 사용한 자원이 반환되지 않을 수 있습니다.

순번	규칙 코드	위험도	규칙 이름
123	0026_RESLEAK	4	부적절한 자원 해제
CWE/CERT		CWE-404,CWE-772	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		142:20 - 143:84	

```
 139                OutputStream outputStream = null;
  140                try {
  141                      outputStream = Files.newOutputStream(Paths.get(zipName));
* 142                     zipOutputSteam = new ZipOutputStream(
  143                              new BufferedOutputStream(new CheckedOutputStream(outputStream,
new Adler32())));
  144                      File topFile = new File(target);
  145                      File[] subFiles = topFile.listFiles();
  146                      if (subFiles != null) {
```

코드 수정 제안	142 행 ZipOutputStream(..) 함수 호출 통해 할당 받은 자원은 사용하지 않는 경우 반환해야 합니다.
	
	142 행 ZipOutputStream(..) 함수 호출 통해 할당 받은 자원은 아래와 유사하게 반환됨이 보장되어야 합니다.
	PrintWriter out = null;
	try {
	out = new PrintWriter(new FileWriter("OutFile.txt"));
	…
	}
	catch (IOException e) { … }
	// 사용한 자원이 반환됨을 보장
	finally { if (out != null) { out.close(); } }

순번	규칙 코드	위험도	규칙 이름
124	0026_RESLEAK	4	부적절한 자원 해제
CWE/CERT		CWE-404,CWE-772	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		143:5 - 143:83	

```
 140            try {
  141                   outputStream = Files.newOutputStream(Paths.get(zipName));
  142                   zipOutputSteam = new ZipOutputStream(
 * 143                           new BufferedOutputStream(new CheckedOutputStream(outputStream,
new Adler32())));
  144                   File topFile = new File(target);
  145                   File[] subFiles = topFile.listFiles();
  146                   if (subFiles != null) {
```

코드 수정 제안	143 행 BufferedOutputStream(..) 함수 호출 통해 할당 받은 자원은 사용하지 않는 경우 반환해야 합니다.
	
	143 행 BufferedOutputStream(..) 함수 호출 통해 할당 받은 자원은 아래와 유사하게 반환됨이 보장되어야 합니다.
	PrintWriter out = null;
	try {
	out = new PrintWriter(new FileWriter("OutFile.txt"));
	...
	}
	catch (IOException e) { ... }
	// 사용한 자원이 반환됨을 보장
	finally { if (out != null) { out.close(); } }

순번	규칙 코드	위험도	규칙 이름
125	0026_RESLEAK	4	부적절한 자원 해제
CWE/CERT		CWE-404,CWE-772	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		143:30 - 143:82	

```
 140            try {
  141                   outputStream = Files.newOutputStream(Paths.get(zipName));
  142                   zipOutputSteam = new ZipOutputStream(
 * 143                           new BufferedOutputStream(new CheckedOutputStream(outputStream,
new Adler32())));
  144                   File topFile = new File(target);
```

```
145                 File[] subFiles = topFile.listFiles();
146                 if (subFiles != null) {
```

코드 수정 제안	143 행 CheckedOutputStream(..) 함수 호출 통해 할당 받은 자원은 사용하지 않는 경우 반환해야 합니다.
	
	143 행 CheckedOutputStream(..) 함수 호출 통해 할당 받은 자원은 아래와 유사하게 반환됨이 보장되어야 합니다.
	PrintWriter out = null;
	try {
	out = new PrintWriter(new FileWriter("OutFile.txt"));
	...
	}
	catch (IOException e) { ... }
	// 사용한 자원이 반환됨을 보장
	finally { if (out != null) { out.close(); } }

순번	규칙 코드	위험도	규칙 이름
126	0026_RESLEAK	4	부적절한 자원 해제
CWE/CERT		CWE-404,CWE-772	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		312:8 - 312:33	

```
309                     return;
310                 }
311
* 312               is = new FileInputStream(file);
313                 byte[] bytes = IOUtils.toByteArray(is);
314                 int pictureIdx = workbook.addPicture(bytes,
HSSFWorkbook.PICTURE_TYPE_JPEG);
```

코드 수정 제안	312 행 FileInputStream(..) 함수 호출 통해 할당 받은 자원은 사용하지 않는 경우 반환해야 합니다.
	
	312 행 FileInputStream(..) 함수 호출 통해 할당 받은 자원은 아래와 유사하게 반환됨이 보장되어야 합니다.
	PrintWriter out = null;
	try {
	out = new PrintWriter(new FileWriter("OutFile.txt"));
	...

	...
	}
	catch (IOException e) { ... }
	// 사용한 자원이 반환됨을 보장
	finally { if (out != null) { out.close(); } }

순번	규칙 코드	위험도	규칙 이름
127	0026_RESLEAK	4	부적절한 자원 해제
CWE/CERT		CWE-404,CWE-772	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/controller/CommonController.java	
발견 위치		70:29 - 70:54	

```
 67          }
  68          response.setCharacterEncoding("UTF-8");
  69          ServletOutputStream out = response.getOutputStream();
* 70          FileInputStream fileIn = new FileInputStream(file);
  71
 72          try {
  73              byte[] buf = new byte[1024];
```

코드 수정 제안	70 행 FileInputStream(..) 함수 호출 통해 할당 받은 자원은 사용하지 않는 경우 반환해야 합니다.
	
	70 행 FileInputStream(..) 함수 호출 통해 할당 받은 자원은 아래와 유사하게 반환됨이 보장되어야 합니다.
	PrintWriter out = null;
	try {
	out = new PrintWriter(new FileWriter("OutFile.txt"));
	...
	}
	catch (IOException e) { ... }
	// 사용한 자원이 반환됨을 보장
	finally { if (out != null) { out.close(); } }

4.11. 잘못된 세션에 의한 데이터 정보 노출 [0029_LEAKSESS]

규칙 코드	0029_LEAKSESS
CWE/CERT	CWE-488
관련 규칙	행안부 SW 보안약점 2021 - 잘못된 세션에 의한 데이터 정보 노출

규칙 설명	다중 스레드 환경에서는 싱글톤(singleton) 객체 필드에 경쟁 조건(Race Condition)이 발생할 수 있습니다. 따라서, 다중 스레드 환경인 java 의 서블릿(servlet) 등에서는 정보를 저장하는 멤버 변수가 포함되지 않도록 하여, 서로 다른 세션에서 데이터를 공유하지 않도록 해야 합니다.

올바른 예제

```
1   <%@page import="javax.xml.namespace.*"%>
2   <%@page import="gov.mogaha.ntis.web.frs.gis.cmm.util.*" %>
3   **<%
4         // JSP 에서 String 필드들이 로컬 변수로 선언되었습니다.
5         String commonPath = "/";
6         String imagePath = commonPath + "img/";
7         String imagePath_gis = imagePath + "gis/cmm/btn/";
8         .....
9   **%>
```

JSP 의 서블릿에 정의한 변수는 _jspService 메소드의 지역변수로 선언되므로 공유가 발생하지 않아 안전합니다.

잘못된 예제

```
1   <%@page import="javax.xml.namespace.*"%>
2   <%@page import="gov.mogaha.ntis.web.frs.gis.cmm.util.*" %>
3   **<%!
4         // JSP 에서 String 필드들이 멤버 변수로 선언됨
5         String username = "/";
6         String imagePath = commonPath + "img/";
7         String imagePath_gis = imagePath + "gis/cmm/btn/";
8         .....
9   **%>
```

JSP 선언부에 선언한 변수는 해당 JSP 에 접근하는 모든 사용자에게 공유됩니다. 먼저 호출한 사용자가 값을 설정하고 사용하기 전에 다른 사용자의 호출이 발생하게 되면, 뒤에 호출한 사용자가 설정한 값이 모든 사용자에게 적용되게 됩니다.

순번	규칙 코드	위험도	규칙 이름
128	0029_LEAKSESS	4	잘못된 세션에 의한 데이터 정보 노출
CWE/CERT		CWE-488	

파일	/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/security/CsrfTokenAdder.java
발견 위치	176:31 - 176:38

```
 173    @Slf4j
  174    class HtmlResponseWrapper extends HttpServletResponseWrapper {
  175
* 176        private ByteArrayOutputStream capture;
  177        private ServletOutputStream output;
  178        private PrintWriter writer;
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
129	0029_LEAKSESS	4	잘못된 세션에 의한 데이터 정보 노출
CWE/CERT		CWE-488	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/security/CsrfTokenAdder.java	
발견 위치		177:29 - 177:35	

```
 174    class HtmlResponseWrapper extends HttpServletResponseWrapper {
  175
 176        private ByteArrayOutputStream capture;
* 177        private ServletOutputStream output;
  178        private PrintWriter writer;
  179
 180        public HtmlResponseWrapper(HttpServletResponse response) throws IOException {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
130	0029_LEAKSESS	4	잘못된 세션에 의한 데이터 정보 노출
CWE/CERT		CWE-488	
파일		/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/security/CsrfTokenAdder.java	
발견 위치		178:21 - 178:27	

```
 175
 176        private ByteArrayOutputStream capture;
  177        private ServletOutputStream output;
* 178        private PrintWriter writer;
```

```
 179
 180        public HtmlResponseWrapper(HttpServletResponse response) throws IOException {
 181                super(response);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
131	0029_LEAKSESS	4	잘못된 세션에 의한 데이터 정보 노출
CWE/CERT		CWE-488	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/security/CsrfTokenAdder.java	
발견 위치		180:29 - 180:35	

```
 177    class HtmlResponseWrapper extends HttpServletResponseWrapper {
 178
 179        private final ByteArrayOutputStream capture;
* 180         private ServletOutputStream output;
 181         private PrintWriter writer;
 182
 183        public HtmlResponseWrapper(HttpServletResponse response) throws IOException {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
132	0029_LEAKSESS	4	잘못된 세션에 의한 데이터 정보 노출
CWE/CERT		CWE-488	
파일		/nmkpg_cyber_user/src/main/java/egovframework/iam/user/security/CsrfTokenAdder.java	
발견 위치		181:21 - 181:27	

```
 178
 179        private final ByteArrayOutputStream capture;
 180         private ServletOutputStream output;
* 181         private PrintWriter writer;
 182
 183        public HtmlResponseWrapper(HttpServletResponse response) throws IOException {
 184                super(response);
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

4.12. 제거되지 않고 남은 디버그 코드 [002A_DEBUGCODE]

규칙 코드	002A_DEBUGCODE
CWE/CERT	CWE-489
관련 규칙	행안부 SW 보안약점 2021 - 제거되지 않고 남은 디버그 코드
규칙 설명	디버깅 목적으로 삽입된 코드는 개발이 완료되면 제거해야 합니다. 디버그 코드는 설정 등의 민감한 정보를 담거나 시스템을 제어하게 허용하는 부분을 담고 있을 수 있습니다. 만일, 남겨진 채로 배포될 경우, 공격자가 식별 과정을 우회하거나 의도하지 않은 정보와 제어 정보가 노출될 수 있습니다.

올바른 예제

```
1  class Base64 {
2        public void otherMethod() { ... }
3  }
```

J2EE 와 같은 응용프로그램에서 main() 메소드는 삭제합니다. J2EE 의 main() 메소드의 경우 디버깅 코드인 경우가 일반적입니다.

잘못된 예제

```
1  class Base64 {
2  **      public static void main(String[] args) {
3  **          if (debug) {
4                  byte[] a = { (byte) 0xfc, (byte) 0x0f, (byte) 0xc0 };
5                  byte[] b = { (byte) 0x03, (byte) 0xf0, (byte) 0x3f };
6                  ......
7              }
8       }
9       public void otherMethod() { ... }
10 }
```

main() 메소드 내의 화면에 출력하는 디버깅 코드를 포함하고 있습니다. J2EE 의 경우 main() 메소드 사용이 필요 없으며, 개발자들이 콘솔 응용프로그램으로 화면에 디버깅코드를 사용하는 경우가 일반적입니다.

순번	규칙 코드	위험도	규칙 이름
133	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/common_popup.js	

발견 위치	124:4 - 124:26

```
 121    function ajaxPost(url, data, callback, showLoading) {
  122        // IE 기본값세팅
  123        showLoading = typeof showLoading !== 'undefined' ? showLoading : true;
* 124        console.log("url", url);
  125        console.log("contextPath", contextPath);
  126        $.ajax({
  127            url: contextPath + url,
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
134	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/common_popup.js	
발견 위치		125:4 - 125:42	

```
 122        // IE 기본값세팅
  123        showLoading = typeof showLoading !== 'undefined' ? showLoading : true;
  124        console.log("url", url);
* 125        console.log("contextPath", contextPath);
  126        $.ajax({
  127            url: contextPath + url,
  128            data: JSON.stringify(data),
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
135	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/bootstrap-datetimepicker/js/bootstrap-datetimepicker.js	
발견 위치		147:16 - 147:138	

```
 144                    this.picker.on({mousewheel: $.proxy(this.mousewheel,this)});
  145                }else
  146                {
* 147                    console.log("Mouse Wheel event is not supported. Please include the
jQuery Mouse Wheel plugin before enabling this option");
```

```
148                     }
149                 }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
136	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/libs/jspdf.min.js	
발견 위치		58:16694 - 58:16728	

```
 55      *
 56      * Licensed under the MIT License. http://opensource.org/licenses/mit-license
 57      */
* 58     ... ept=a}},pushMask:function(){var t="function"==typeof
this.pdf.internal.newObject2;if(!t)return void console.log("jsPDF v2 not enabled");var
e=this.pdf.internal.newStreamObject(),n=this.pdf.internal.newObject2();n.push("<</Type /ExtGSta ...
 59      * jsPDF fromHTML plugin. BETA stage. API subject to change. Needs browser
 60      * Copyright (c) 2012 Willow Systems Corporation, willow-systems.com
 61      *                         2014 Juan Pablo Gaviria, https://github.com/juanpgaviria
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
137	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/pages/login.js	
발견 위치		31:3 - 31:28	

```
 28                  form.id.value = id;
 29                  form.password.value = "ENC|" + pw;
 30
* 31                 console.log(form.id.value);
 32                  console.log(form.password.value);
 33
 34                  form.submit();
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름

138	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/pages/login.js	
발견 위치		32:3 - 32:34	

```
 29                  form.password.value = "ENC|" + pw;
  30
 31                  console.log(form.id.value);
* 32                   console.log(form.password.value);
  33
 34                  form.submit();
 35              }
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
139	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_admin/src/main/webapp/assets/js/pages/newsroom/form.js	
발견 위치		65:6 - 65:14	

```
 62
 63          var $form = $('#form-newsroom');
  64          $('#btn-save').on('click', function () {
* 65              debugger;
  66
 67              if($form.valid()){
  68                  $form.submit();
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
140	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/common.js	
발견 위치		171:4 - 171:26	

```
 168    function ajaxPost(url, data, callback, showLoading) {
  169         // IE 기본값세팅
  170         showLoading = typeof showLoading !== 'undefined' ? showLoading : true;
```

```
* 171          console.log("url", url);
  172          console.log("contextPath", contextPath);
  173          $.ajax({
  174                url: contextPath + url,
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
141	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/common.js	
발견 위치		172:4 - 172:42	

```
  169          // IE 기본값세팅
  170          showLoading = typeof showLoading !== 'undefined' ? showLoading : true;
  171          console.log("url", url);
* 172          console.log("contextPath", contextPath);
  173          $.ajax({
  174                url: contextPath + url,
  175                data: JSON.stringify(data),
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
142	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/common_popup.js	
발견 위치		124:4 - 124:26	

```
  121    function ajaxPost(url, data, callback, showLoading) {
  122          // IE 기본값세팅
  123          showLoading = typeof showLoading !== 'undefined' ? showLoading : true;
* 124          console.log("url", url);
  125          console.log("contextPath", contextPath);
  126          $.ajax({
  127                url: contextPath + url,
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름

143	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/common_popup.js	
발견 위치		125:4 - 125:42	

```
 122        // IE 기본값세팅
  123        showLoading = typeof showLoading !== 'undefined' ? showLoading : true;
  124        console.log("url", url);
* 125        console.log("contextPath", contextPath);
  126        $.ajax({
  127            url: contextPath + url,
  128            data: JSON.stringify(data),
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
144	002A_DEBUGCODE	4	제거되지 않고 남은 디버그 코드
CWE/CERT		CWE-489	
파일		/nmkpg_cyber_user/src/main/webapp/assets/js/pages/user/join_signup.js	
발견 위치		89:3 - 89:40	

```
 86
 87          $('#btn-checked-id').on('click', function() {
  88              var id = $('#id').val().trim();
* 89              console.log($('#checkedIdFlag').val());
  90
 91              if (id === '') {
  92                  alert('아이디를 입력하세요');
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

4.13. Public 메소드로부터 반환된 Private 배열 [002C_RETPRIVA]

규칙 코드	002C_RETPRIVA
CWE/CERT	CWE-495
관련 규칙	행안부 SW 보안약점 2021 - Public 메소드로부터 반환된 Private 배열
규칙 설명	private 로 선언된 배열을 public 으로 선언된 메소드로 반환(return)하면, 그 배열의 레퍼런스가 외부에 공개되어 외부에서 배열수정과 객체 속성변경이 가능해집니다.

올바른 예제

```
1   private Color[] colors;
 2  // 메소드를 private 으로 하거나, 복제본 반환, 수정하는 public 메소드를 별도로 만듭니다.
 3  public void onCreate(Bundle savedInstanceState) {
 4        super.onCreate(savedInstanceState);
 5        Color[] newColors = getUserColors();
 6        ......
 7  }
 8  public Color[] getUserColors(Color[] userColors) {
 9        // 배열을 복사합니다.
10  **      Color[] colors = new Color [userColors.length];
11  **      for (int i = 0; i  <  colors.length; i++)
12              // clone()메소드를 이용하여 배열의 원소도 복사합니다.
13  **            colors[i] = this.colors[i].clone();
14  **      return colors;
15  }
```

private 배열에 대한 복사본을 만들고, 복사된 배열의 원소로는 clone() 메소드로 private 배열의 원소의 복사본을 만들어 저장하여 반환하도록 작성하면, private 선언된 배열과 원소에 대한 의도하지 않은 수정을 방지할 수 있습니다.

잘못된 예제

```
1   // private 인 배열을 public 인 메소드가 return 합니다.
 2  **private Color[] colors;
 3  **public Color[] getUserColors(Color[] userColors) { return colors; }
```

멤버 변수 colors 는 private 로 선언되었지만 public 으로 선언된 getColors() 메소드로 참조를 얻을 수 있습니다. 이 경우 의도하지 않은 수정이 발생할 수 있습니다.

순번	규칙 코드	위험도	규칙 이름
145	002C_RETPRIVA	3	Public 메소드로부터 반환된 Private 배열
CWE/CERT		CWE-495	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelParser.java	
발견 위치		235:2 - 235:14	

```
232                 rows[i].parse(obj);
 233                 i++;
 234             }
```

```
* 235             return rows;
  236         }
  237
  238         @SuppressWarnings("unchecked")
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
146	002C_RETPRIVA	3	Public 메소드로부터 반환된 Private 배열
CWE/CERT		CWE-495	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelParser.java	
발견 위치		249:2 - 249:14	

```
  246                 rows[i].parse(obj);
  247                 i++;
  248             }
* 249             return rows;
  250         }
  251
  252         public int[] getWidths() {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

순번	규칙 코드	위험도	규칙 이름
147	002C_RETPRIVA	3	Public 메소드로부터 반환된 Private 배열
CWE/CERT		CWE-495	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelParser.java	
발견 위치		253:2 - 253:16	

```
  250         }
  251
  252         public int[] getWidths() {
* 253             return widths;
  254         }
  255
  256         public String getProfile() {
```

코드 수정 제안	적합한 수정제안이 존재하지 않습니다.

4.14. 오류메시지를 통한 정보노출 [7003_LEAKERRORMSG]

규칙 코드	7003_LEAKERRORMSG
CWE/CERT	CWE-209,CWE-497
관련 규칙	행안부 SW 보안약점 2021 - 오류 메시지 정보노출
규칙 설명	오류메시지나 스택정보에 시스템 내부구조가 포함되어 민감한 정보, 디버깅 정보가 노출 가능합니다.

올바른 예제

```
1   try {
 2         rd = new BufferedReader(new FileReader(new File(filename)));
 3   } catch(IOException e) {
 4         // 에러 코드와 정보를 별도로 정의하고 최소 정보만 로깅
 5   **     logger.error("ERROR-01: 파일 열기 에러");
 6       } finally {
 7             try {
 8                   rd.close();
 9              } catch (IOException ex) {
10   **                 logger.error("ERROR-02: 파일 닫기 에러");
11                  }
12              }
```

예외 이름이나 오류추적 정보를 출력하지 않도록 합니다.

잘못된 예제

```
1   try {
 2         rd = new BufferedReader(new FileReader(new File(filename)));
 3   } catch(IOException e) {
 4   // 에러 메시지를 통해 스택 정보가 노출됨
 5   **       e.printStackTrace();
 6   }
```

오류메시지에 예외 이름이나 오류추적 정보를 출력하여 프로그램 내부 정보가 유출되는 경우이다.

순번	규칙 코드	위험도	규칙 이름
148	7003_LEAKERRORMSG	3	오류메시지를 통한 정보노출
CWE/CERT		CWE-209,CWE-497	

파일	/nmkpg_cyber_admin/src/main/java/egovframework/iam/admin/controller/BoardFeedbackController.java
발견 위치	107:13 - 107:27

```
 104                     ExcelExporter.export(outputStream, list.toArray(), "feedback",
 105                                                 HEADERS, PROPERTIES, COLUMN_WIDTHS, commonProperty,
new ArrayList<>());
 106                 } catch (IOException e) {
* 107                     log.error(e.getMessage(), e);
 108                 }
 109
 110         }
```

코드 수정 제안	107행 e.getMessage 를 사용하면 프로그램 내부구조를 쉽게 파악할 수 있습니다. 107행 e.getMessage 사용하지 마세요. 디버깅을 위해 필요한 경우 개발 과정에서만 확인할 수 있게 로그로 남기세요.

순번	규칙 코드	위험도	규칙 이름
149	7003_LEAKERRORMSG	3	오류메시지를 통한 정보노출
CWE/CERT		CWE-209,CWE-497	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/FileUtil.java	
발견 위치		390:13 - 390:27	

```
 387                     FileUtils.forceDelete(file);
 388                 }
 389             } catch (IOException e) {
* 390                 log.error(e.getMessage());
 391             }
 392         }
```

코드 수정 제안	390행 e.getMessage 를 사용하면 프로그램 내부구조를 쉽게 파악할 수 있습니다. 390행 e.getMessage 사용하지 마세요. 디버깅을 위해 필요한 경우 개발 과정에서만 확인할 수 있게 로그로 남기세요.

순번	규칙 코드	위험도	규칙 이름
150	7003_LEAKERRORMSG	3	오류메시지를 통한 정보노출
CWE/CERT		CWE-209,CWE-497	

파일	/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelExporter.java
발견 위치	134:16 - 134:30

```
131
132             workbook.write(outputStream);
133           } catch (IOException e) {
* 134             log.error(e.getMessage(), e);
135           } finally {
136             try {
137                 outputStream.close();
```

코드 수정 제안	134행 e.getMessage 를 사용하면 프로그램 내부구조를 쉽게 파악할 수 있습니다. 134행 e.getMessage 사용하지 마세요. 디버깅을 위해 필요한 경우 개발 과정에서만 확인할 수 있게 로그로 남기세요.

순번	규칙 코드	위험도	규칙 이름
151	7003_LEAKERRORMSG	3	오류메시지를 통한 정보노출
CWE/CERT		CWE-209,CWE-497	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelExporter.java	
발견 위치		139:18 - 139:32	

```
136             try {
137                 outputStream.close();
138             } catch (IOException e) {
* 139                 log.error(e.getMessage(), e);
140             }
141         }
142     }
```

코드 수정 제안	139행 e.getMessage 를 사용하면 프로그램 내부구조를 쉽게 파악할 수 있습니다. 139행 e.getMessage 사용하지 마세요. 디버깅을 위해 필요한 경우 개발 과정에서만 확인할 수 있게 로그로 남기세요.

순번	규칙 코드	위험도	규칙 이름
152	7003_LEAKERRORMSG	3	오류메시지를 통한 정보노출

CWE/CERT	CWE-209,CWE-497
파일	/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java
발견 위치	82:3 - 82:22

```
 79                     rowsPrint(parser);
  80                     outputExcel(parser, response, request);
  81                 } catch (Throwable e) {
* 82                     e.printStackTrace();
  83                 }
  84             }
```

코드 수정 제안	82행 e.printStackTrace 를 사용하면 프로그램 내부구조를 쉽게 파악할 수 있습니다. 82행 e.printStackTrace 사용하지 마세요. 디버깅을 위해 필요한 경우 개발 과정에서만 확인할 수 있게 로그로 남기세요.

순번	규칙 코드	위험도	규칙 이름
153	7003_LEAKERRORMSG	3	오류메시지를 통한 정보노출
CWE/CERT		CWE-209,CWE-497	
파일		/nmkpg_cyber_core/src/main/java/com/stoneitgt/util/excel/ExcelWriter.java	
발견 위치		342:4 - 342:25	

```
 339                         if (is != null)
  340                             is.close();
  341                     } catch (IOException ioe) {
* 342                         ioe.printStackTrace();
  343                     }
  344                 }
  345             }
```

코드 수정 제안	342행 ioe.printStackTrace 를 사용하면 프로그램 내부구조를 쉽게 파악할 수 있습니다. 342행 ioe.printStackTrace 사용하지 마세요. 디버깅을 위해 필요한 경우 개발 과정에서만 확인할 수 있게 로그로 남기세요.

순번	규칙 코드	위험도	규칙 이름
154	7003_LEAKERRORMSG	3	오류메시지를 통한 정보노출

CWE/CERT	CWE-209,CWE-497
파일	/nmkpg_cyber_user/src/main/java/egovframework/iam/user/filters/JwtAuthenticationFilter.java
발견 위치	129:22 - 129:36

```
126             try {
127                     out = httpServletResponse.getWriter();
128              } catch (IOException e) {
* 129                   log.error(e.getMessage(), e);
130              }
131
132             if (out != null) {
```

코드 수정 제안	129행 e.getMessage 를 사용하면 프로그램 내부구조를 쉽게 파악할 수 있습니다. 129행 e.getMessage 사용하지 마세요. 디버깅을 위해 필요한 경우 개발 과정에서만 확인할 수 있게 로그로 남기세요.

붙임1 소스코드 보안약점 진단항목

□ 입력데이터 검증 및 표현

○ 프로그램 입력 값에 대한 검증 누락 또는 부적절한 검증, 데이터의 잘못된 형식지정으로 인해 발생할 수 있는 보안약점

번호	보안약점	설 명
1	SQL 삽입	SQL 질의문을 생성할 때 검증되지 않은 외부 입력 값을 허용하여 악의적인 질의문이 실행가능한 보안약점
2	코드 삽입	프로세스가 외부 입력 값을 코드(명령어)로 해석·실행할 수 있고 프로세스에 검증되지 않은 외부 입력 값을 허용한 경우 악의적인 코드가 실행 가능한 보안약점
3	경로 조작 및 자원 삽입	시스템 자원 접근경로 또는 자원제어 명령어에 검증되지 않은 외부 입력값을 허용하여 시스템 자원에 무단 접근 및 악의적인 행위가 가능한 보안약점
4	크로스사이트 스크립트	사용자 브라우저에 검증되지 않은 외부 입력값을 허용하여 악의적인 스크립트가 실행가능한 보안약점
5	운영체제 명령어 삽입	운영체제 명령어를 생성할 때 검증되지 않은 외부 입력값을 허용하여 악의적인 명령어가 실행 가능한 보안약점
6	위험한 형식 파일 업로드	파일의 확장자 등 파일형식에 대한 검증없이 파일 업로드를 허용하여 공격이 가능한 보안약점
7	신뢰되지 않는 URL주소로 자동접속 연결	URL 링크 생성에 검증되지 않은 외부 입력값을 허용하여 악의적인 사이트로 자동 접속 가능한 보안약점
8	부적절한 XML 외부개체 참조	임의로 조작된 XML 외부개체에 대한 적절한 검증 없이 참조를 허용하여 공격이 가능한 보안약점
9	XML 삽입	XQuery, XPath 질의문을 생성할 때 검증되지 않은 외부 입력값을 허용하여 악의적인 질의문이 실행가능한 보안약점
10	LDAP 삽입	LDAP 명령문을 생성할 때 검증되지 않은 외부 입력 값을 허용하여 악의적인 명령어가 실행가능한 보안약점
11	크로스사이트 요청 위조	사용자 브라우저에 검증되지 않은 외부 입력 값을 허용하여 사용자 본인의 의지와는 무관하게 공격자가 의도한 행위가 실행 가능한 보안약점
12	서버사이드 요청 위조	서버 간 처리되는 요청에 검증되지 않은 외부 입력값을 허용하여 공격자가 의도한 서버로 전송하거나 변조하는 보안약점
13	HTTP 응답분할	HTTP 응답헤더에 개행문자(CR이나 LF)가 포함된 검증되지 않은 외부 입력값을 허용하여 악의적인 코드가 실행 가능한 보안약점
14	정수형 오버플로우	정수형 변수에 저장된 값이 허용된 정수 값 범위를 벗어나 프로그램이 예기치 않게 동작 가능한 보안약점
15	보안기능 결정에 사용되는 부적절한 입력값	보안기능(인증, 권한부여 등) 결정에 검증되지 않은 외부 입력값을 허용하여 보안기능을 우회하는 보안약점
16	메모리 버퍼 오버플로우	메모리 버퍼의 경계값을 넘어서 메모리값을 읽거나 저장하여 예기치 않은 결과가 발생하는 보안약점
17	포맷 스트링 삽입	printf 등 포맷 스트링 제어함수에 검증되지 않은 외부 입력값을 허용하여 발생하는 보안약점

□ 보안기능

○ 보안기능(인증, 접근제어, 기밀성, 암호화, 권한 관리 등)을 부적절하게 구현 시 발생할 수 있는 보안약점

번호	보안약점	설 명
1	적절한 인증 없는 중요기능 허용	중요정보(금융정보, 개인정보, 인증정보 등)를 적절한 인증없이 열람(또는 변경) 가능한 보안약점
2	부적절한 인가	중요자원에 접근할 때 적절한 제어가 없어 비인가자의 접근이 가능한 보안약점
3	중요한 자원에 대한 잘못된 권한 설정	중요자원에 적절한 접근 권한을 부여하지 않아 중요정보가 노출·수정 가능한 보안약점
4	취약한 암호화 알고리즘 사용	중요정보(금융정보, 개인정보, 인증정보 등)의 기밀성을 보장할 수 없는 취약한 암호화 알고리즘을 사용하여 정보가 노출 가능한 보안약점
5	암호화되지 않은 중요 정보	중요정보(비밀번호, 개인정보 등) 전송 시 암호화 또는 안전한 통신채널을 이용하지 않거나, 저장 시 암호화하지 않아 정보가 노출 가능한 보안약점
6	하드코드된 중요정보	소스코드에 중요정보(비밀번호, 암호화키 등)를 직접 코딩하여 소스코드 유출 시 중요정보가 노출되고 주기적 변경이 어려운 보안약점
7	충분하지 않은 키 길이 사용	암호화 등에 사용되는 키의 길이가 충분하지 않아 데이터의 기밀성·무결성을 보장할 수 없는 보안약점
8	적절하지 않은 난수 값 사용	사용한 난수가 예측 가능하여, 공격자가 다음 난수를 예상해서 시스템을 공격 가능한 보안약점
9	취약한 비밀번호 허용	비밀번호 조합규칙(영문, 숫자, 특수문자 등) 미흡 및 길이가 충분하지 않아 비밀번호가 노출 가능한 보안약점
10	부적절한 전자서명 확인	프로그램, 라이브러리, 코드의 전자서명에 대한 유효성 검증이 적절하지 않아 공격자의 악의적인 코드가 실행 가능한 보안약점
11	부적절한 인증서 유효성 검증	인증서에 대한 유효성 검증이 적절하지 않아 발생하는 보안약점
12	사용자 하드디스크에 저장되는 쿠키를 통한 정보 노출	쿠키(세션 ID, 사용자 권한정보 등 중요정보)를 사용자 하드디스크에 저장되어 중요정보가 노출 가능한 보안약점
13	주석문 안에 포함된 시스템 주요정보	소스코드 주석문에 인증정보 등 시스템 주요정보가 포함되어 소스코드 노출 시 주요정보도 노출 가능한 보안약점
14	솔트 없이 일방향 해쉬함수 사용	솔트를 사용하지 않고 생성된 해쉬 값으로부터 공격자가 미리 계산된 레인보우 테이블을 이용하여 해쉬 적용 이전 원본 정보를 복원가능한 보안약점
15	무결성 검사 없는 코드 다운로드	소스코드 또는 실행파일을 무결성 검사 없이 다운로드 받아 실행하는 경우, 공격자의 악의적인 코드가 실행 가능한 보안약점
16	반복된 인증시도 제한 기능 부재	인증 시도 수를 제한하지 않아 공격자가 반복적으로 임의 값을 입력하여 계정 권한을 획득 가능한 보안약점

□ 시간 및 상태

○ 동시 또는 거의 동시 수행을 지원하는 병렬시스템, 하나 이상의 프로세스가 동작되는 환경에서 시간 및 상태를 부적절하게 관리하여 발생할 수 있는 보안약점

번호	보안약점	설 명
1	경쟁조건: 검사 시점과 사용 시점(TOCTOU)	멀티 프로세스 상에서 자원을 검사하는 시점과 사용하는 시점이 달라서 발생하는 보안약점
2	종료되지 않는 반복문 또는 재귀 함수	종료조건 없는 제어문 사용으로 반복문 또는 재귀함수가 무한히 반복되어 발생할 수 있는 보안약점

□ 에러처리

○ 에러를 처리하지 않거나, 불충분하게 처리하여 에러 값에 중요 정보(시스템 등)가 포함될 때 발생할 수 있는 보안약점

번호	보안약점	설 명
1	오류 메시지 정보노출	오류메시지나 스택정보에 시스템 내부구조가 포함되어 민감한 정보, 디버깅 정보가 노출 가능한 보안약점
2	오류 상황 대응 부재	시스템 오류상황을 처리하지 않아 프로그램 실행정지 등 의도하지 않은 상황이 발생 가능한 보안약점
3	부적절한 예외 처리	예외사항을 부적절하게 처리하여 의도하지 않은 상황이 발생 가능한 보안약점

□ 코드오류

○ 타입변환 오류, 자원(메모리 등)의 부적절한 반환 등과 같이 개발자가 범할 수 있는 코딩오류로 인해 유발되는 보안약점

번호	보안약점	설 명
1	Null Pointer 역참조	변수의 주소 값이 Null인 객체를 참조하는 보안약점
2	부적절한 자원 해제	사용 완료된 자원을 해제하지 않아 자원이 고갈되어 새로운 입력을 처리할 수 없는 보안약점
3	해제된 자원 사용	메모리 등 해제된 자원을 참조하여 예기치 않은 오류가 발생하는 보안약점
4	초기화되지 않은 변수 사용	변수를 초기화하지 않고 사용하여 예기치 않은 오류가 발생하는 보안약점
5	신뢰할 수 없는 데이터의 역직렬화	악의적인 코드가 삽입·수정된 직렬화 데이터를 적절한 검증 없이 역직렬화하여 발생하는 보안약점

□ 캡슐화

○ 중요한 데이터 또는 기능성을 불충분하게 캡슐화 하였을 때, 인가되지 않은 사용자에게 데이터 누출이 가능해지는 보안약점

번호	보안약점	설 명
1	잘못된 세션에 의한 데이터 정보 노출	잘못된 세션에 의해 인가되지 않은 사용자에게 중요정보가 노출 가능한 보안약점
2	제거되지 않고 남은 디버그 코드	디버깅을 위한 코드를 제거하지 않아 인가되지 않은 사용자에게 중요정보가 노출 가능한 보안약점
3	Public 메소드부터 반환된 Private 배열	Public으로 선언된 메소드에서 Private로 선언된 배열을 반환(return)하면 Private 배열의 주소 값이 외부에 노출되어 해당 Private 배열값을 외부에서 수정 가능한 보안약점
4	Private 배열에 Public 데이터 할당	Public으로 선언된 데이터 또는 메소드의 인자가 Private으로 선언된 배열에 저장되면 이 Private배열을 외부에서 접근하여 수정 가능한 보안약점

□ API 오용

○ 의도된 사용에 반하는 방법으로 API를 사용하거나, 보안에 취약한 API를 사용하여 발생할 수 있는 보안약점

번호	보안약점	설 명
1	DNS lookup에 의존한 보안결정	도메인명 확인(DNS lookup)으로 보안결정을 수행할 때 악의적으로 변조된 DNS 정보로 예기치 않은 보안위협에 노출되는 보안약점
2	취약한 API 사용	취약한 함수를 사용해서 예기치 않은 보안위협에 노출되는 보안약점

붙임2	서비스 범위 정의서

□ 5-8. 웹 보안 클리닉

1 정 의	
서비스 정의(설명)	○ 관리원 내에서 운영 중이거나 새로 개발된 홈페이지에 대해 수동 및 자동 진단 도구를 통해 보안취약점을 점검하고 입주기관에 결과를 제공하며 취약점 제거를 위한 기술지원 및 이행점검 지원 ○ 홈페이지에 대한 보안취약점이 발생하지 않도록 개발 시 지켜야 할 가이드라인을 개발 및 제공
서비스 제공 부서	○ **디지털안전상황실**
2 역할 및 책임	
전제사항	○ 홈페이지 보안취약점 및 소스코드 보안약점에 대한 점검, 제거 등의 관리주체는 입주기관이며, 관리원은 점검 및 제거 지원 ○ 관리원의 운영환경 및 여건의 범위 내에서 지원
관리원	○ 홈페이지 개발 시 지켜야 할 가이드라인을 입주기관에 제공 ○ 홈페이지의 보안취약점 점검 또는 모의해킹(PT : Penetration Test)을 년 1회 이상 실시 ○ 보안 취약점 제거를 위한 기술지원 및 이행점검 지원 ○ 관리원은 입주기관이 보안취약점 및 소스코드 보안약점 점검 요청 시 수시로 점검을 수행하고, 결과 제공
입주기관	○ 홈페이지 개발 시 관리원에서 제공하는 가이드라인 및 행정안전부 소프트웨어 개발보안 가이드를 준수 ○ 운영 중인 홈페이지 관련자료(URL, OS, WEB/WAS·DB 등)를 관리원에 제공 ○ 입주기관 책임 하에 보안취약점 점검·제거 등 추진 - 관리원에서 점검결과를 제공하는 경우 보안취약점 제거 ○ 신규 또는 개편되는 홈페이지에 대해서는 보안취약점 점검을 요청하고 제거 후 서비스 개시
공동	○ 없음
3 서비스 상세 사항	
서비스 범위	○ **소스코드 보안취약점 점검 후 입주기관 통보** ○ **취약점 제거 관련 기술 및 이행점검 지원**
서비스 시간	○ **서비스 제공시간 : 근무일 기준 09:00~18:00**
서비스 측정 지표	○ **소스코드 보안취약점 양호율, 기관요청처리 만족도**