## Q1

**[보기]는 스마트의료 분야의 보안위협을 설명하고 있다.
어떤 위협에 대한 것인지 가장 알맞은 것을 고르시오.**

[보기]

- **디버그 포트를 이용한 펌웨어 획득** : 개발 시 사용된 디버그 포트를 제거하지 않아 펌웨어 등을 획득하는 공격으로 공격자가 내부 소스코드 및 구조를 파악 할 수 있으며, 이를 기반으로 알려지지 않은 취약점을 확인하거나 특정 부분을 변조하여 주입공격 등을 할 수 있다.
- **부채널 공격** : 전송되는 정보에 대한 암호 알고리즘이 작동할 때 전기 소모량, 전자기 신호량 등 을 분석해서 암호키 등을 유추 할 수 있다.

**[답가지]**

① 스마트의료 기기 보안위협
② 스마트의료 게이트웨이 보안위협
③ 스마트의료 네트워크 보안위협
④ 스마트의료 정보시스템 보안위협



## Q2

**[보기]는 사이버 공격 사례를 설명하고 있다.
어떤 사이버 공격의 사례인지 올바른 것을 고르시오.**

[보기]

- 워너크라이(WannaCry) : 2017년 5월 12일 전세계 150여개 국에서 최소 30만대 이상의 컴퓨터 시스템들이 감염. 감염된 시스템은 특정 확장자를 가지는 내부 파일들이 .WNCRY로 변경되고 파일 내용이 암호화되며, 감염시스템 화면에 안내문구를 표시한다.
- 클롭(Clop) : 2019년 3월 러시아 해킹그룹에 의해 제작됨, 국내 증권사 직원 PC가 감염되어 전산장애 발생. 파일을 암호화하고 확장자는 .Clop으로 변경된다.
- 갠드크랩(GandCrab) : 2018년 1월 처음 등장하여 전문지식 없이도 공격 가능한 서비스 형으로 제작이 가능하여 공격 증가의 주요 원인이 되었음. 파일을 암호화 하고 확장자는 .GDCB, .KRAB등으로 변경된다.

**[답가지]**

① 피싱(Phishing)
② 파밍(Pharming)
③ 랜섬웨어(Ransomware)
④ DDoS(Distributed Denial of Service)

## Q3 다음 중 무선 랜(Wireless LAN)의 기술적 보안 취약점에 해당하지 않는 것은?

[답가지]

① 도청

② 서비스 거부

③ 불법 AP(Rouge AP)

④ 전파관리 수준의 미흡

4. 정보보안 이해와 활용 단답형 Q4

## Q4 다음 보기에서 설명하는 보안 기술을 무엇이라 하는가?

[보기]

- 주로 공공기관이나 기업에서 인터넷과 완전히 격리된 환경에서 업무를 볼 수 있도록 내부 네트워크를 분리하는 기술

- 해당 기술에는 논리적인 방법과 물리적인 방법이 있음

5. 정보보안 이해와 활용 수행형 Q5

# Q5

김 대리가 수행하는 프로젝트에서는 개발 언어로 JAVA를 사용하며, 기본적으로 시큐어 코딩을 적용하여야 한다. [보기]는 외부의 입력을 통하여 "디렉터리 경로 문자열"을 생성하여 특정 처리를 하는 코드로, 시큐어 코딩을 적용하지 않은 상태이다. 다음 물음에 답 하시오. 단, 시큐어 코딩 조치 과정에서 상대 경로를 지정하는데 사용하지 않아야 할 문자는 "/" 한 종류만 고려하기로 한다.

1) [보기 1]의 코드에서 보안에 취약한 라인 번호를 제시하고 그 이유를 설명 하시오.

2) [보기 2]의 코드에서 각 취약점을 해결하기 위해 취해야 할 시큐어 코딩 조치를 박스 위치의 라인에 적용 하시오.

[보기 1]은 외부로부터 파일 명을 입력 받고 그 앞에 상대 디렉터리 경로를 추가하여 해당 파일 객체를 생성하고 그 파일을 삭제하는 코드의 예이다. 각 명령 줄에 라인 번호를 편의 상 부여하였다.

참고 : Properties 클래스는 주로 어플리케이션의 환경 설정과 관련된 속성을 저장하는데 사용되며 데이터를 파일로부터 읽고 기록하는 편리한 기능을 제공한다.

[보기 1]

```
1: ……
2: public void f(Properties request) {
3:     ……
4:     String name = request.getProperty("filename");
5:     if(name != null && !"".equals(name)) {
6:         File file = new File("/usr/local/tmp/" + name);
7:         file.delete();
8:     }
9:     ……
10: }
```

[보기 2]는 [보기 1]에 대해 시큐어 코딩 조치를 취하기 위하여 완성하려는 코드이다.

[보기 2]

```
1: ……
2: public void f(Properties request) {
3:     ……
4:     String name = request.getProperty("filename");
5:     if (name != null && !"".equals(name)) {
6:         [                                    ];
7 :        File file = new File("/usr/local/tmp/" + name);
8:         [                                    ];
9:     }
10: ……
11: }
```