

[SW보안 개론- 2분반] 6월 17일, 첫 번째 단기 진행과제 (총 105점, 5점은 보너스)

다음 문제들을 꼼꼼하게 읽고 보고서를 작성하시오.

1. 최신 스마트 냉장고는 인터넷에 연결되어 사용되고 있어, connected refrigerator라고 한다. 사이버 위협 관점에서, internet-connected refrigerator가 일반 데스크톱 PC보다 더 위험한 이유를 자세히 설명하시오. (5점)
2. Internet-connected refrigerator를 보안 위협들로부터 안전하게 보호하기 위해서 취할 수 있는 방안(행위)들을 모두 나열하고 자세히 설명하시오. (5점)
3. 인터넷 뱅킹 서비스에서 발생할 수 있는 위협을 STRIDE 모델과 연관지어 설명하시오. 또한, 이들 위협을 어떤 보안 기법(메커니즘)으로 방어(또는 예방)할 수 있는지를 설명하시오. (10점)
4. Passive attack과 Active attack의 예를 들고, 또한 차이점을 STRIDE와 관련 지어 설명하시오. (6점)
5. 사용자 인증 방식으로 (1) ID/패스워드 방식, (2) 생체인증 방식(지문이나 홍채를 사용한 방식)을 사용할 수 있다. 두 가지 기법의 장단점을 서로 비교하여 설명하시오. (6점)
6. 한국에 있는 길동이(송신자)가 제작한 2GB 영화를 미국의 Bob(수신자)에게 암호화하여 안전하고 효율적으로 전송하고자 한다. 둘은 먼저 어떤 암호화 방식(들)을 사용할 것인지를 정한 다음, 진행해야 한다. 어떤 암호화 방식(들)을 사용하여 어떻게 전송하는 것이 안전하면서 효율적인지를 설명하시오. 송/수신자가 선택한 방식을 사용하여 안전하고 효율적으로 영화를 전송하는 과정을 단계별로 상세하게 설명하시오. (15점)
7. Diffie-Hellman알고리즘과 RSA알고리즘을 비교 설명하시오. 각 알고리즘의 역할 상의 차이점은? (6점)
8. Hill cipher를 사용하여, "time"을 암호화 하시오. 또한, 암호화된 암호문을 복호화 하시오. (대소문자 구별이 없으며, 단,  $a=0, b=1, c=2, \dots$ ). 암호화 키 행렬이 아래와 같을 때, 암호화 및 복호화 하는 과정을 자세히 설명하시오. (15점)

12	3
20	7

9. Double Transposition을 사용하여 아래 평문(원문)을 암호화하시오. 중간 과정도 같이 보이시오. 단, Column Key는 "Keyword" 이고, Row Key는 "matrix"임. 블록 단위로 처리하기 바라며, 필요하다면 추가 가정을 해도 됩니다. (5점)  
(평문) "Remember that Knowledge in youth is wisdom in age"
10. 4 개의 rail을 가진 rail fence cipher가 사용될 때 다음 물음에 답하시오. 단, 1번 rail부터 시작하고 key=4라고 가정하며 대소문자 구별 없습니다. 필요하다면 추가 가정을 하면 됩니다. 다음 암호문을 복호화 하시오. (7점)

(암호문) **TSAT NHIN YLDO RIEE OORA LANR ROEG**

11. 갑순이와 같은 학과에 30명(갑순이 포함)의 사람이 있을 때, 갑순이와 같은 생일을 갖는 사람이 있을 확률은? 계산식만 정확하게 작성해도 됩니다. 이 문제는 암호학적 해시 함수가 가져야 할 어떤 특성과 관련이 가장 많은가요? 그 이유를 설명하시오. (10점)
12. 공인인증서 갱신 기간이 필요한 이유는? 공인인증서를 유지하자는 의견과 공인인증서를 폐지하자는 의견이 있었습니다. 어느 쪽 의견에 찬성하나요? 그 이유를 자세히 설명하시오. (15점)

[SW보안 개론- 3분반] 6월 17일, 첫 번째 단기 진행과제 (총 105점, 5점은 보너스)

다음 문제를 읽고 물음에 구체적이고 상세하게 보고서를 작성하시오.

1. 피싱(phishing)의 유형을 2가지 이상 설명하고, 이러한 피싱 방지 방법에 대해 자세히 설명하시오. (10점)
2. 해킹된 컴퓨터는 어떻게 악용될 수 있는지를 서로 다른 방식 5가지 이상 설명하시오. (10점)
3. 안드로이드 앱에 대해, 어떠한 보안 위협이 존재할 수 있는지 3가지 이상 설명하시오. 이 보안 위협을 방어하는 기법에 대해 자세히 설명하시오. (10점)
4. 한국에 있는 길동(송신자)이가 한국에서 작성된 1GB 서류를 전자 서명(digital signature)하여 미국의 Bob(수신자)에게 효율적이면서 안전하게 전송하고자 한다. 둘은 먼저 어떤 알고리즘들을 어떤 순서로 사용할 것인지를 정한 다음, 진행해야 한다. 어떤 알고리즘들을 사용하여 무엇을 어떻게 전송하는 것이 효율적이고 안전한지를 설명하시오. 선택한 방식으로 효율적으로 전송하는 과정을 단계별로 나누어 자세히 설명하시오. (15점)
5. Mono-alphabetic substitution cipher와 Poly-alphabetic substitution cipher 중에서 무엇이 더 안전한가? 그 이유는? (5점)
6. DES와 SEED 암호 기법(암호 알고리즘)을 비교하는 표를 작성하시오. 어느 알고리즘이 더 안전한가? 그 이유는? (10점)
7. 패스워드를 AES로 암호화하여 저장하는 방법과 SHA-256으로 해시화하여 저장하는 방법 중에서 어느 방법이 안전한지 설명하시오. 그 이유를 자세히 설명하시오. (10점)
8. Hill cipher를 사용하여, "word"를 암호화 하시오. 또한, 암호화 된 암호문을 복호화 하시오. (대소문자 구별이 없으며, 단,  $a=0, b=1, c=2, \dots$ ). 암호화 키 행렬이 아래와 같을 때, 암호화 및 복호화 하는 과정을 자세히 설명하시오. (15점)

8	15
12	24

9. 4 개의 rail을 가진 rail fence cipher가 사용될 때 다음 물음에 답하시오. 단, 1번 rail부터 시작하며  $key=4$ 라고 가정하며 대소문자 구별 없습니다. 필요하다고 판단되면 추가 가정을 하면 됩니다. 다음 평문을 암호화 하시오. (5점)

(평문) **Seeing is believing**

10. 6개의 column을 기반으로 하는 Columnar transposition이 있다. 암호복호화에 사용되는 키워드는 "friend"이다. 다음 암호문에 대한 평문은? 그 과정을 보이시오. 블록 단위로 처리하는 것은 권장합니다. (5점)

(암호문) **LRNUCWVWYAYLKTUEOLYIEUOBONNIR**

11. 한 학급에 20명의 사용자가 있을 때, 생일이 같은 사람이 존재할 확률은? 계산식만 자세하게 작성해도 좋습니다. 이 문제는 암호학적 해시 함수가 가져야 할 어떤 특성과 관련이 가장 많은가요? 그 이유를 설명하시오. (10점)

## [SW보안 개론- 2분반] 6월 24일, 두 번째 단기 진행과제

다음 물음에 번호 순으로 답하시오. (총 11문제 110점) 풀이과정이나 원리 설명이 더 중요합니다.

13:00~14:10분까지 보고서 정리하고, 14:15분까지 PDF로 생성하여 email로 제출해야 합니다.

1. char, short 자료형의 범위가 오른쪽과 같을 때, 다음 프로그램들의 출력 결과를 보이고, 왜 그러한 결과가 나오는지에 대해 상세히 설명하시오. 출력 결과 자체 보다는 왜 그러한 결과가 나오는지를 설명하는 것이 훨씬 더 중요 합니다. (12점)

```
#include <stdio.h> // my_char_3.c
```

```
void main()
```

```
{  
    char c1, c2, csub, csum;
```

```
    c1 = 100; c2 = 128;
```

```
    csub = c1-c2;    csum = c1+c2;
```

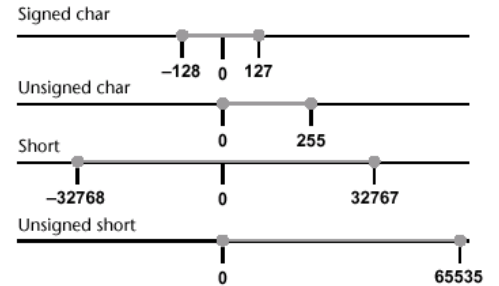
```
    printf("%hd(0hx%hx), %hd(0hx%hx) \n", c1, c1, c2, c2);
```

```
    printf("%hhd(0hhx%hx, %hhd(0hhx%hx) \n", c1, c1, c2, c2);
```

```
    printf("%hd(0hx%hx), %hd(0hx%hx) \n", csub, csub, csum, csum);
```

```
    printf("%hhd(0hhx%hx), %hd(0hhx%hx) \n", csub, csub, csum,  
csum);
```

```
}
```



2. 다음 프로그램의 출력 결과를 보이고, 왜 그러한 결과가 나오는지에 대해 상세히 설명하시오. (13점)

```
include <stdio.h> // my_short_2.c
```

```
void main()
```

```
{
```

```
    int j = 0xcafe6789;
```

```
    unsigned short k = 0x7FFF;
```

```
    short m = k;
```

```
    short n = 0x8000;
```

```
    char c;
```

```
    printf("%d(0x%x), %d(0x%x)\n", k, k, m, m);
```

```
    printf("%d(0x%x), %d(0x%x)\n", k+2, k+2, m+2, m+2);
```

```
    printf("%hu(0hx%hx), %hd(0hx%hx)\n", k+2, k+2, m+2, m+2);
```

```
    m = n - 0xFFFF;
```

```
    printf("n - 0xFFFF = %hd(0hx%hx) \n", m, m);
```

```
    m = j; c = j;
```

```
    printf("m = %hd(0hx%hx), c = %hhd(0hhx%hx) \n", m, m, c, c);
```

```
}
```

시간이 부족하면 10진수는 양수인지 음수인지 설명만 해도 됩니다. 단, 16진수 출력은 정확하게 보이고, 16진수에 대응되는 10진수가 양수인지 음수인지 밝히시오.

왜 그렇게 출력되는지를 함께 설명하시오.

3. 다음 프로그램 수행 후, 출력되는 값은? (5점)

<pre>#include &lt;stdio.h&gt; // my_arr_2.c void main() {     unsigned short k;     char array[5] = {'a', 'a', 'a', 'a', 'a'};      for(k=1; k&lt;=5; k++)         array[k] = 'a'+k;     array[k] = 'B';     for(int i=0; i&lt;=k; i++)         printf("%2c ", array[i]);     printf("\n"); }</pre>	<pre>\$ gcc -o my_arr_2 my_arr_2.c \$ ./my_arr_2</pre> <p>수행 결과는 →</p>
---	--

4. stack\_guard.c를 아래와 작성한 다음, 인텔 64-bit 구조의 Ubuntu에서 컴파일하여, stack\_guard0를 생성하였다. 버퍼 오버플로우 버그를 악용하여 'you win!'을 출력할 수 있게 하는 gets()의 입력을 보여야 합니다. python을 사용한 입력 값과 파이프라인()을 사용하시오. (10점)

\$ gcc -fno-stack-protector -o stack\_guard0 stack\_guard.c

<pre>#include &lt;stdio.h&gt; /* stack_guard.c */ #include &lt;string.h&gt; #define goodPass "GOODPASS"  int main () {     char passIsGood = '0';     short canary = 40;     int canary2 = 0xff;     char buf[38];      printf("%08x, %x, %x, %08x\n", buf, &amp;canary2, &amp;canary, &amp;passIsGood);     printf("Enter password: \n");     gets(buf);      printf("canary2 = 0x%x, canary= 0x%x\n", canary2, canary);      if(canary != 40    canary2 != 0xff){         printf("BOF attack!\n");         return(-1);     }     if(strcmp (buf, goodPass)==0) passIsGood = '3';     if(passIsGood == '3')         printf("you win!\n");     return 0; }</pre>	<p>참고: 주소는 다음과 같다.</p> <p>&amp;passIsGood: 0xc96af1ff &amp;canary: 0xc96af1fc &amp;canary2: 0xc96af1f8 Buf: 0xc96af1d0</p> <p>입력의 총 길이는 몇 바이트이어야 하는지도 설명하시오.</p>
--	--

5. 삼성전자나 Naver, Google 등이 software bug bounty program을 운영하는 이유는? (5점)

6. Microsoft SDL에서 Static analysis와 Dynamic analysis의 차이점, 그리고 장단점에 자세히 설명하시오. Fuzz testing에 대해서도 설명하시오. (10점)

7. SW보안에서 penetrate and patch 접근방법이란? 이 접근방법의 문제점에 대해 자세히 설명하시오? 또한 이 문제를 해결하기 위한 방안도 설명하시오. (10점)

8. 다음 프로그램의 실행 결과를 보이시오. (15점)

<pre>#include &lt;stdio.h&gt; // arrays_3.c void main() {     short *ptr;     int a[5] = {0, 1, 2, 3, 4};     int b[6] = {5, 6, 7, 8, 9, 10};     int c[7] = {11, 12, 13, 14, 15, 16, 17};     void print_arr(int *, unsigned short);      printf("a= 0x%x, b=0x%x, c=0x%x\n", a, b, c);     ptr = (short *)b;     ptr[-12] = 100;          ptr[-8] = 200;     ptr[-4] = 300;           ptr[0] = 400;     ptr[4] = 500;            ptr[8] = 600;     ptr[12] = 700;           ptr[16] = 800;     ptr[18] = 900;           ptr[20] = 1000;     print_arr(a, 5);     print_arr(b, 6);     print_arr(c, 7); }  void print_arr(int *arr, unsigned short k) {     for(int i=0; i&lt;k; i++)         printf("%4d ", arr[i]);     printf("\n"); }</pre>	<pre>\$ gcc -o arrays_3 arrays_3.c \$ ./arrays_3</pre> <p>a[]의 시작 주소: 0x5d23b120 b[]의 시작주소: 0x5d23b140 c[]의 시작주소: 0x5d23b160 a= 0x5d23b120, b=0x5d23b140, c=0x5d23b160</p> <p>ptr의 타입은 "short *" 입니다. 출력을 보이고, 왜 그렇게 출력되는지 그 이유도 함께 설명하시오.</p>
--	--

9. OWASP Top 10 Vulnerabilities 중에서 Injection에 대해 설명하시오. Injection과 관련된 CWE를 4개 이상 나열하여 함께 설명하시오. (10점)

10. 컴퓨터 웜과 Zombies (Bots)을 특징을 설명하고 비교하시오. (5점)

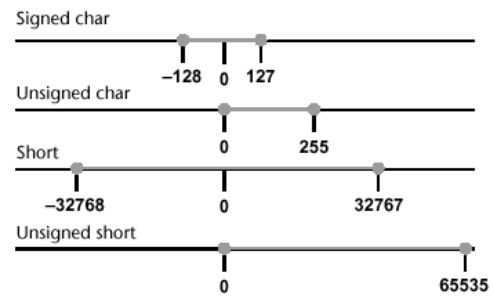
11. Malware 은닉 기법을 5가지 나열하고, 각각 자세하게 설명하시오. (15점)

## [SW보안 개론- 3분반] 6월 24일, 두 번째 단기 진행과제

다음 물음에 번호 순으로 답하시오. (총 12문제 110점) 풀이과정이나 원리 설명이 더 중요합니다.

10:30~11:40분까지 보고서 정리하고, 11:45분까지 PDF로 생성하여 email로 제출해야 합니다.

1. Software bug와 software vulnerability의 차이점은? (5점)
2. char 와 short 자료형들의 범위가 오른쪽과 같을 때, 다음 프로그램들의 출력 결과를 보이고, 왜 그러한 결과가 나오는지에 대해 상세히 설명하시오. 출력 결과 자체 보다는 왜 그러한 결과가 나오는지를 설명하는 것이 훨씬 더 중요 합니다. (10점)

<pre>#include &lt;stdio.h&gt; // my_char_2.c  void main() {     unsigned short us1 = 65535;     short s1 = 0x7FFF;     char c = -1;     int r1, r2;      r1 = r2 = 0;     if(c == us1)         r1 = printf("Why is -1 == 65535 ???\n");     if(c &lt; us1)         r2 = printf("Why is -1 &lt; 65535 ???\n");     printf("0x%x, 0x%x\n", c, us1);     printf("%d, %d\n", r1, r2);     printf("s1 + 1 = %d, %hd, 0x%x\n", s1+1, s1+1, s1+1); }</pre>	 <p>The diagram illustrates the memory ranges for different C data types:</p> <ul style="list-style-type: none"> <li><b>Signed char:</b> Range from -128 to 127.</li> <li><b>Unsigned char:</b> Range from 0 to 255.</li> <li><b>Short:</b> Range from -32768 to 32767.</li> <li><b>Unsigned short:</b> Range from 0 to 65535.</li> </ul>
---	---

3. 다음 프로그램의 출력 결과를 보이고, 왜 그러한 결과가 나오는지에 대해 상세히 설명하시오. (15점)

<pre>#include &lt;stdio.h&gt; // my_short_3.c  void main() {     int j = 0xbeefcafe;     unsigned short k = 0xFFFF;     short m = k;     short n = 0x800;     char c;      printf("%d(0x%x), %d(0x%x)\n", k, k, m, m);     printf("%d(0x%x), %d(0x%x)\n", k+1, k+1, m+1, m+1);     printf("%hd(0x%x), %hd(0x%x)\n", k+1, k+1, m+1, m+1);     printf("%hd(0x%x), %hd(0x%x)\n", k-1, k-1, m-1, m-1);      m = 2*n;     printf("2 * n = %hd(0x%x) \n", m, m);      m = j; c = j;     printf("m = %d(0x%x), c = %d(0x%x) \n", m, m, c, c); }</pre>	<p>앞 5개의 printf()문에서는 10진수와 16진수 출력을 정확하게 보이시오.</p> <p>단, 마지막 printf()문에서 m의 10진수 값은 양수인지 음수인지 설명만 해도 됩니다. (16진수 출력은 정확하게 보이고, 16진수로 출력되는 숫자가 양수인지 음수인지 밝히시오.)</p>
--	--

4. 다음 프로그램의 출력 결과는? 그 이유도 같이 설명하시오. (5점)

```
#include <stdio.h>
void main()
{
    int k = 1;          int val = 0;

    while (k = 10) {
        k++;            val++;
        if(val > 20) break;
    }
    printf("k = %d, val = %d\n", k, val);
}
```

5. stack\_guard.c를 아래와 작성한 다음, 인텔 64-bit 구조의 Ubuntu에서 컴파일하여, stack\_guard0를 생성하였다. 버퍼 오버플로우 버그를 악용하여 'you win!'을 출력할 수 있게 하는 gets()의 입력을 보여야 합니다. python을 사용한 입력 값과 파이프라인()을 사용하시오. (10점)

\$ gcc -fno-stack-protector -o stack\_guard0 stack\_guard.c

```
#include <stdio.h> /* stack_guard.c */
#include <string.h>
#define goodPass "GOODPASS"
int main () {
    char passIsGood = 0;
    short canary = 35;
    char canary2 = '3';
    char buf[32];
    printf("%08x, %x, %x, %08x\n", buf, &canary2, &canary, &passIsGood);
    printf("Enter password: \n");
    gets(buf);
    if(canary != 35 || canary2 != '3'){
        printf("BOF attack!\n");
        return(-1);
    }
    if(strcmp (buf, goodPass)==0) passIsGood =5;
    if(passIsGood == 5)
        printf("you win!\n");
    return 0;
}
```

참고:

지역변수의 주소는 다음과 같다고 가정한다.

&passIsGood: 0x9c096eef  
&canary: 0x9c096eec  
&canary2: 0x9c096eeb  
Buf: 0x9c096ec0

입력의 총 길이는 몇 바이트이어야 하는지도 설명하시오.

6. 다음 물음에 답하시오. (8점)

우측 프로그램("my\_cp.c") 프로그램이 왜 보 안에 취약한지 설명하시오.

\$ make my\_cp

위와 같이 컴파일 한 후에, 실제 실행 예를 보이시오. 즉, 아래와 같이 실행할 때, 000 와 ???에 어떤 값들이 주어질 때 위험한지 구체적인 예를 보이시오.

\$ ./my\_cp 000 ???

```
1. #include <stdio.h> /* my_cp.c */
2. #include <stdlib.h>
3.
4. int main(int argc, char *argv[]) {
5.     int retval;
6.     char buffer[50];
7.
8.     if (argc < 3) {
9.         printf("usage: my_cp src_file dst_file\n");
10.        return -1;
11.    }
12.    sprintf(buffer, "cp %s %s", argv[1], argv[2]);
13.    system(buffer);
14. }
```

7. 다음 프로그램의 실행 결과를 보이시오. (17점)

<pre>#include &lt;stdio.h&gt; // arrays_3.c void main() {     int *ptr;     int a[5] = {0, 1, 2, 3, 4};     int b[6] = {5, 6, 7, 8, 9, 10};     int c[7] = {11, 12, 13, 14, 15, 16, 17};     void print_arr(int *, unsigned short);      printf("a= 0x%x, b=0x%x, c=0x%x\n", a, b, c);     ptr = b+1;     ptr[-6] = 100;          ptr[-4] = 200;     ptr[-2] = 300;          ptr[0] = 400;     ptr[2] = 500;           ptr[4] = 600;     ptr[6] = 700;           ptr[8] = 800;     ptr[10] = 900;     print_arr(a, 5);     print_arr(b, 6);     print_arr(c, 7); } void print_arr(int *arr, unsigned short k) {     for(int i=0; i&lt;k; i++)         printf("%4d ", arr[i]);     printf("\n"); }</pre>	<pre>\$ gcc -o arrays_3 arrays_3.c \$ ./arrays_3  a[]의 시작 주소: 0xe9cea5e0 b[]의 시작주소: 0xe9cea600 c[]의 시작주소: 0xe9cea620 a= 0xe9cea5e0, b=0xe9cea600, c=0xe9cea620 ptr에는 b+1 가 할당됩니다.  출력을 보이고, 왜 그렇게 출력되는지 그 이유도 함께 설명하시오.</pre>
---	---

8. Secure SDLC에서 요구사항 분석에는 security requirements를 포함한다. Good password에 대한 security requirements를 3개 이상 설명하시오. (6점)

9. Microsoft SDL에서 Threat modeling이란 무엇인지 자세히 설명하시오. (10점)

10. 컴퓨터 바이러스와 컴퓨터 웜의 차이점을 표를 그려 설명하시오. (6점)

11. Ransomware와 Cryptojacking malware에 대해 설명하고, 차이점을 자세하게 설명하시오. (이전 homework에서 나온 문제임) (8점)

12. Malware analysis에서 Static analysis와 Dynamic analysis의 차이점, 그리고 장단점에 대해 자세히 설명하시오. (10점)