Who hijacked My Smart Home

--A url hacked all IOT devices

Han Zidong@tencent

Self Introduction

- Han Zidong
  - Android security researcher from Tencent Mobile Security Lab
  - Focus on mobile security research, especially App vulnerability and IOT related security research

Agenda

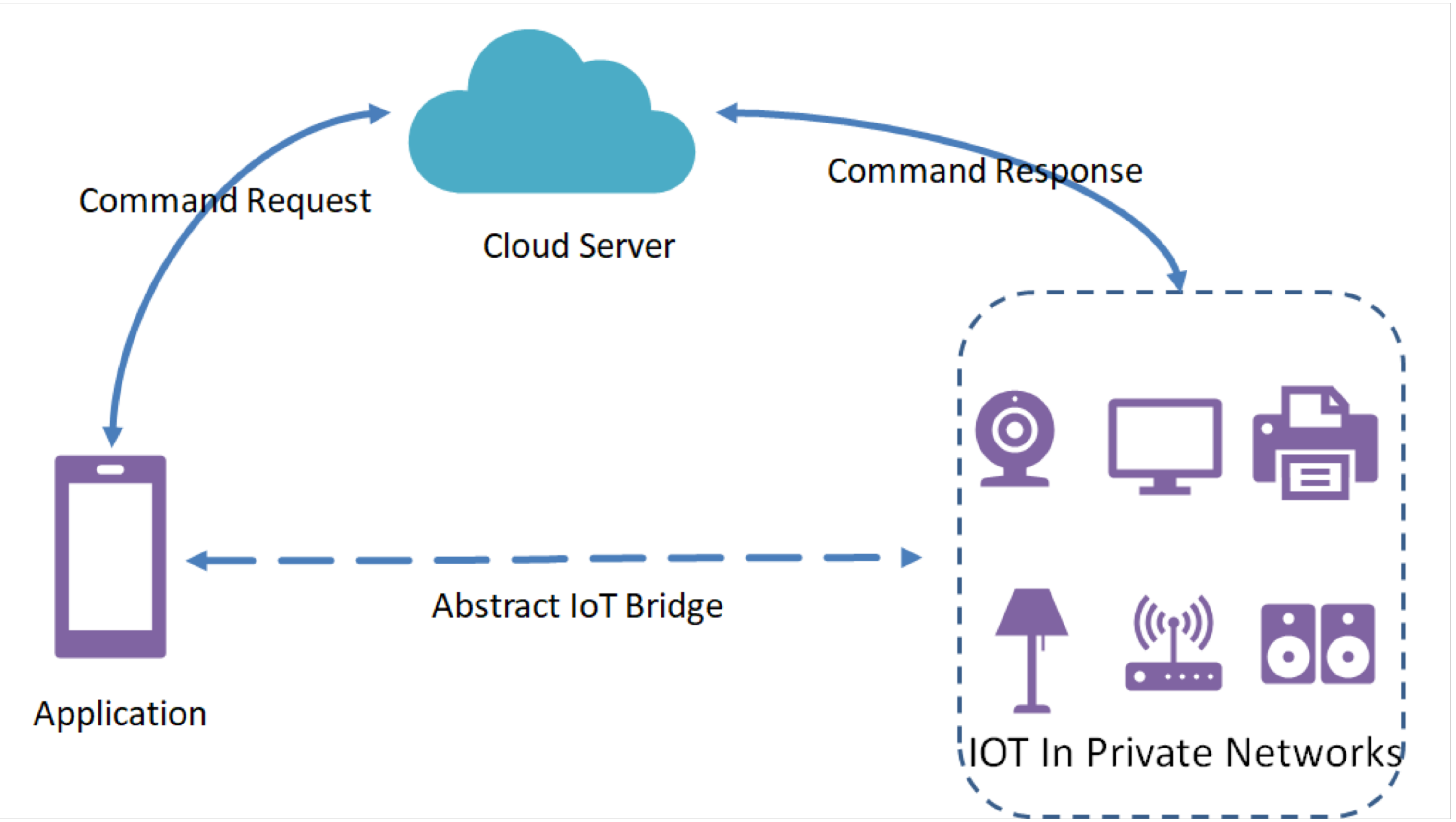
- ➢ Smart home security introduction
- ➢ Tradtional attack in IoT
- ➢ Our advanced approach
- ➢ Case of some vulnerabilities
- ➢ Q&A

Smart home security introduction

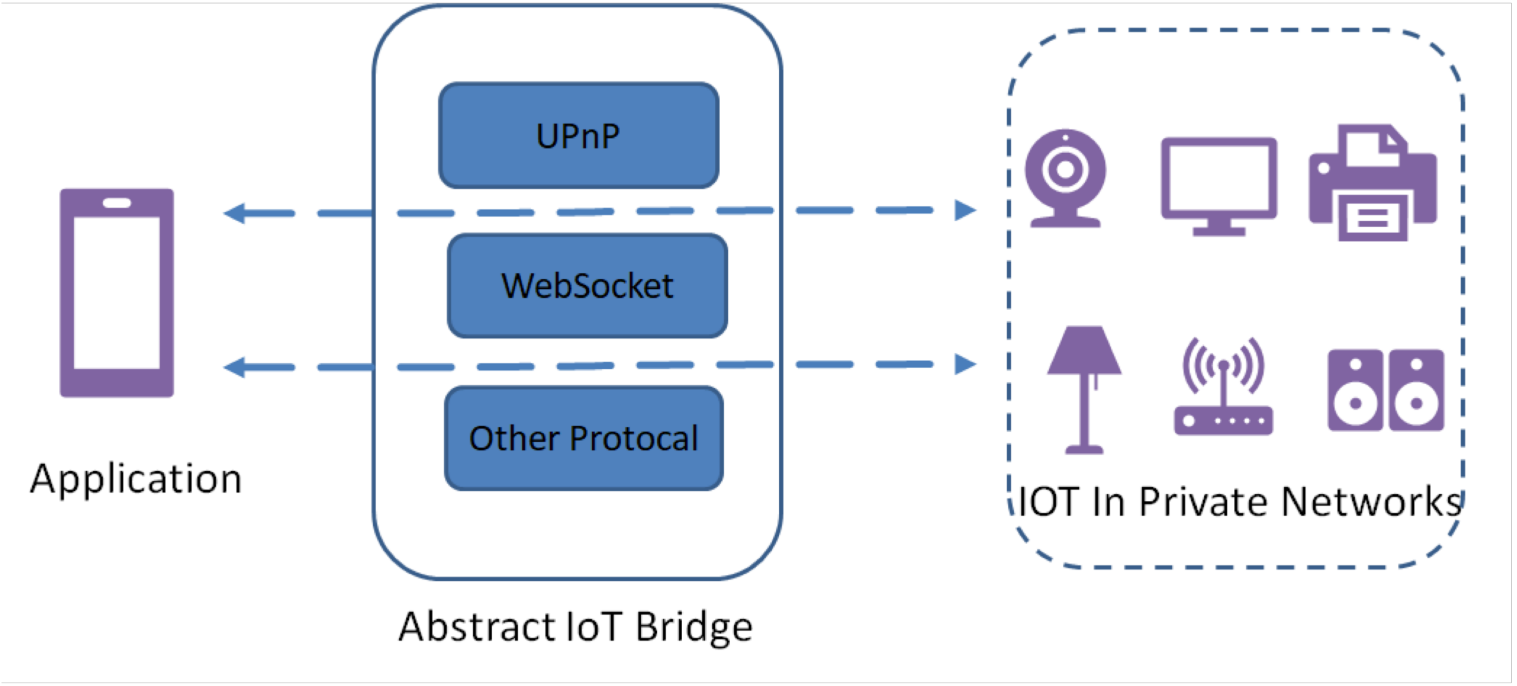
- Smart home architecture

- Vulnerability in IoT Device

IoTBridge With Cloud Server



IoTBridge Without Cloud Server



Vulnerability in IoT Device

- Security in Smart Home
  - More and More IoT device (Smart Tv, lock,router,robot .etc…)
  - What makes security risks in Smart Device
- IoT Vulnerability
  - The characteristic of "Internet of Everything" makes convenience of hacking
  - Something bridges IoT with App in an insecure way

Vulnerability in IoT Device

- What do we do?
  - Analyze every risks of smart home
  - Hack IoT device in an advanced approach
  - Attack from only a Url and gain control during a short time

Tradtional attack in IoT

- Attack target device
  - Single point attack in IoT devices with more intelligent action
    - Smart Tv ,Smart Router,Smart Speaker and etc…
  - Combined Attack in IoT devices with gateway dependency
    - Smart lamp,Smart adapt,Smart cleaner robot,Smart lock and etc…

Tradtional attack in IoT

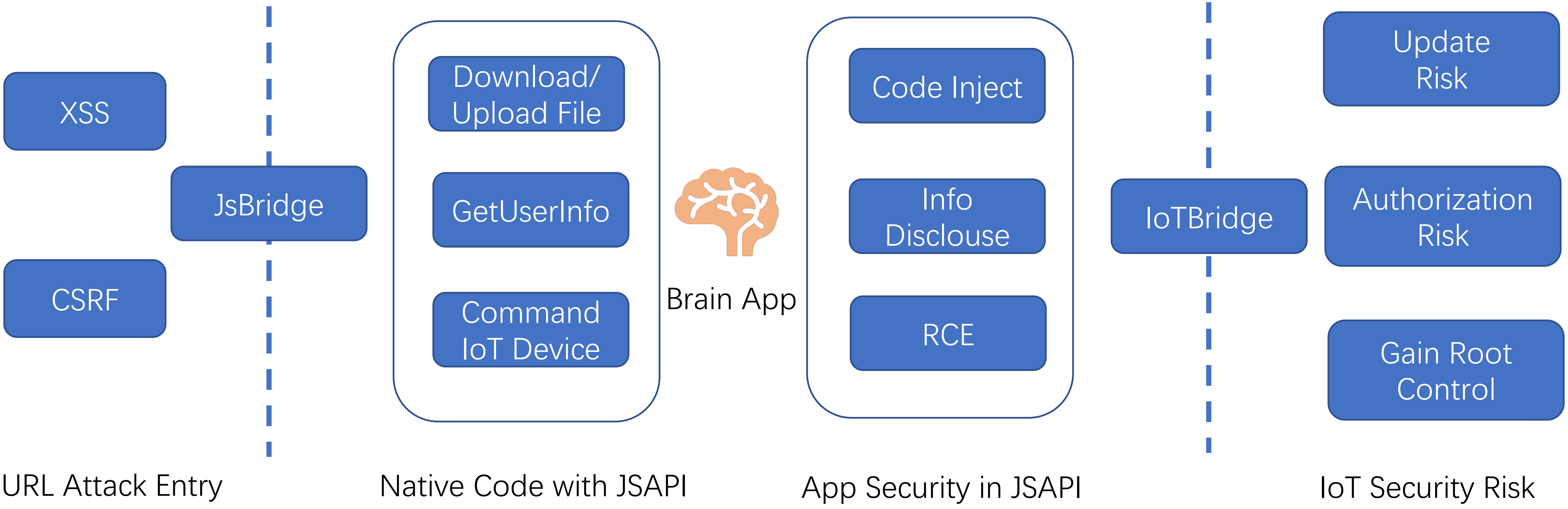
- Common Attack Approach
  - Heap or Stack Overflow attack
  - Command Inject
  - Android/Linux N-Day CVE
  - External IP and sensitive interface exposure

Some New Attack Approach

- Why a url?
  - As a trap to attack more concealed
- What can a url do?
  - Gain control of IoT in some way
- Some Attack Surface
  - Attack IoT bridge protocol
  - Security in brain App of IoT
  - More …

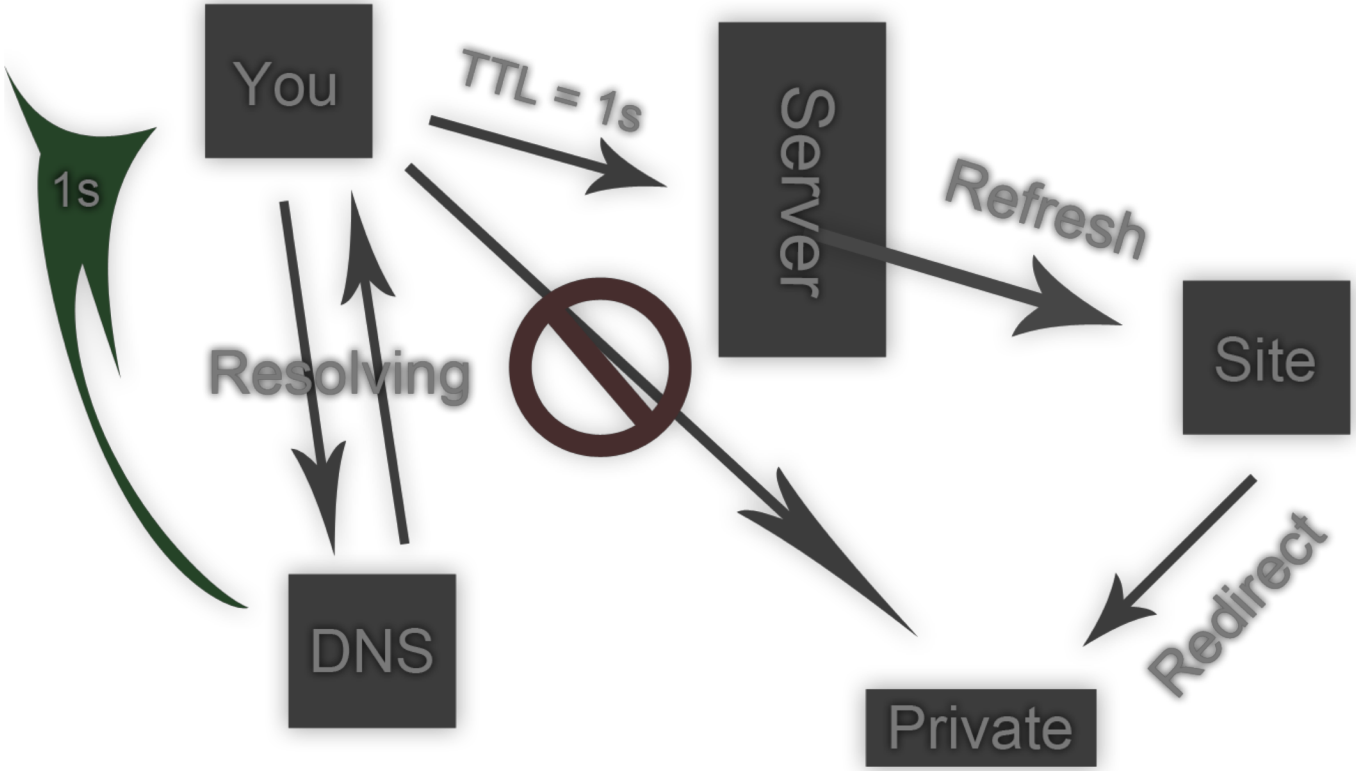
Expand ability of a remote url

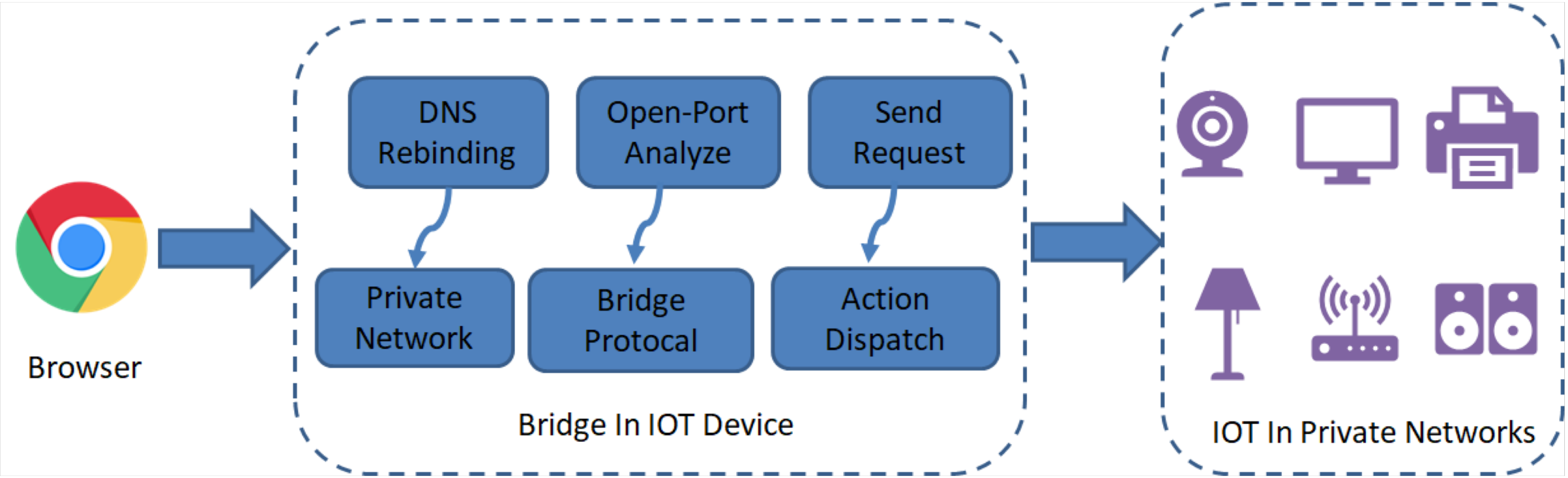
- How to combine app and IoT Security
  - Expolit JSAPI of brain app



Expand ability of a remote url

- Csrf and penetrate into private net
  - Dns-Rebinding



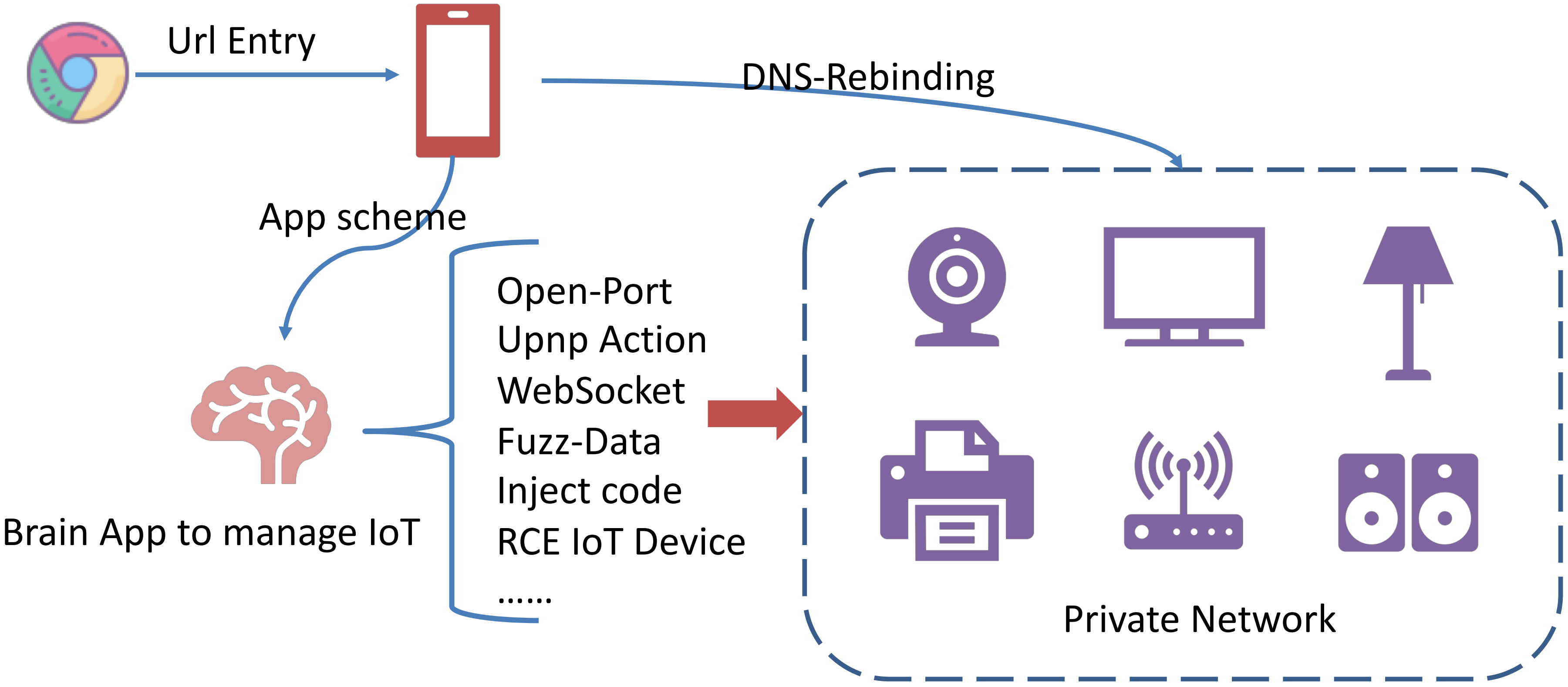


Url Attack Smart Home

Advanced target to Attack

- How to make attack more persistence and concealed
  - More intelligent ,more chance
    - ➢Smart TV
    - ➢Smart Speaker
    - ➢Smart Router
  - Better attack approach to gain control
    - suddenly playing a horror film
    - Silent install the backdoor
    - Samsung Tv turned off in a fake way to record user’ s voice

RCE From RemoteUrl



Url Entry

DNS-Rebinding

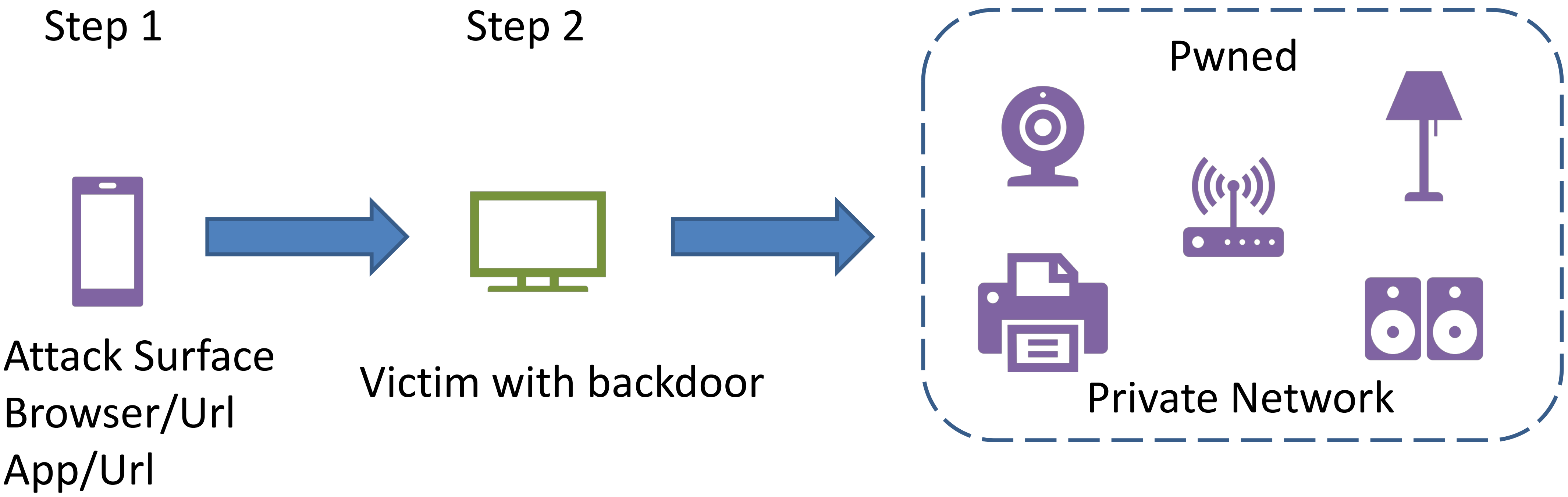
App scheme

Brain App to manage IoT

Open-Port
Upnp Action
WebSocket
Fuzz-Data
Inject code
RCE IoT Device
……

Private Network

- Attack open-port to get protocal type
- Analyze sensitive action or exposed interface
- Inject backdoor to access persisting RCE attack



Step 1

Step 2

Pwned

Attack Surface
Browser/Url
App/Url

Victim with backdoor

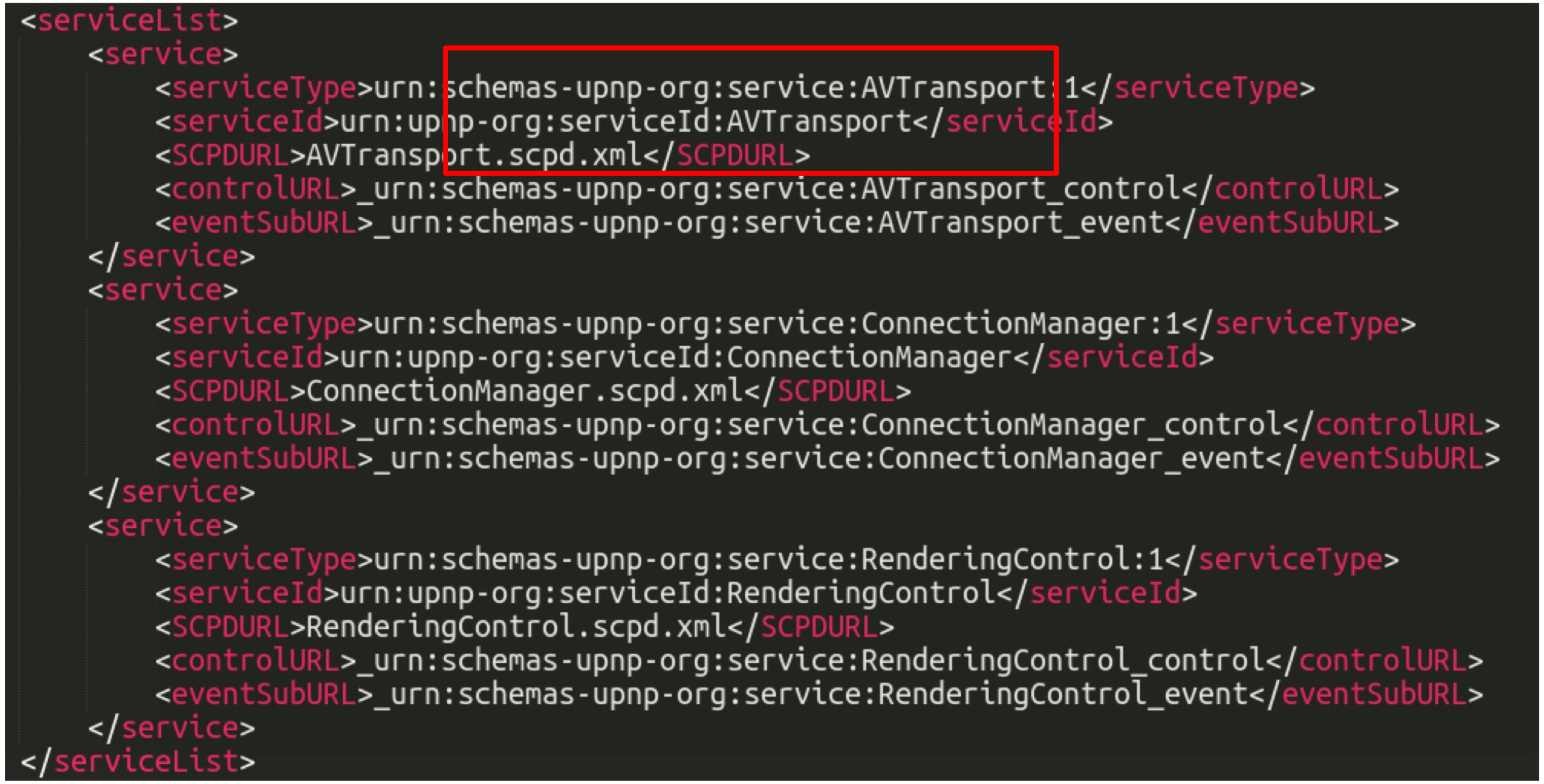
Private Network

Smart Tv = Backdoor?

Cases Study

- ➢ Smart Tv attack case
- ➢ GeekPwn hacker-house case

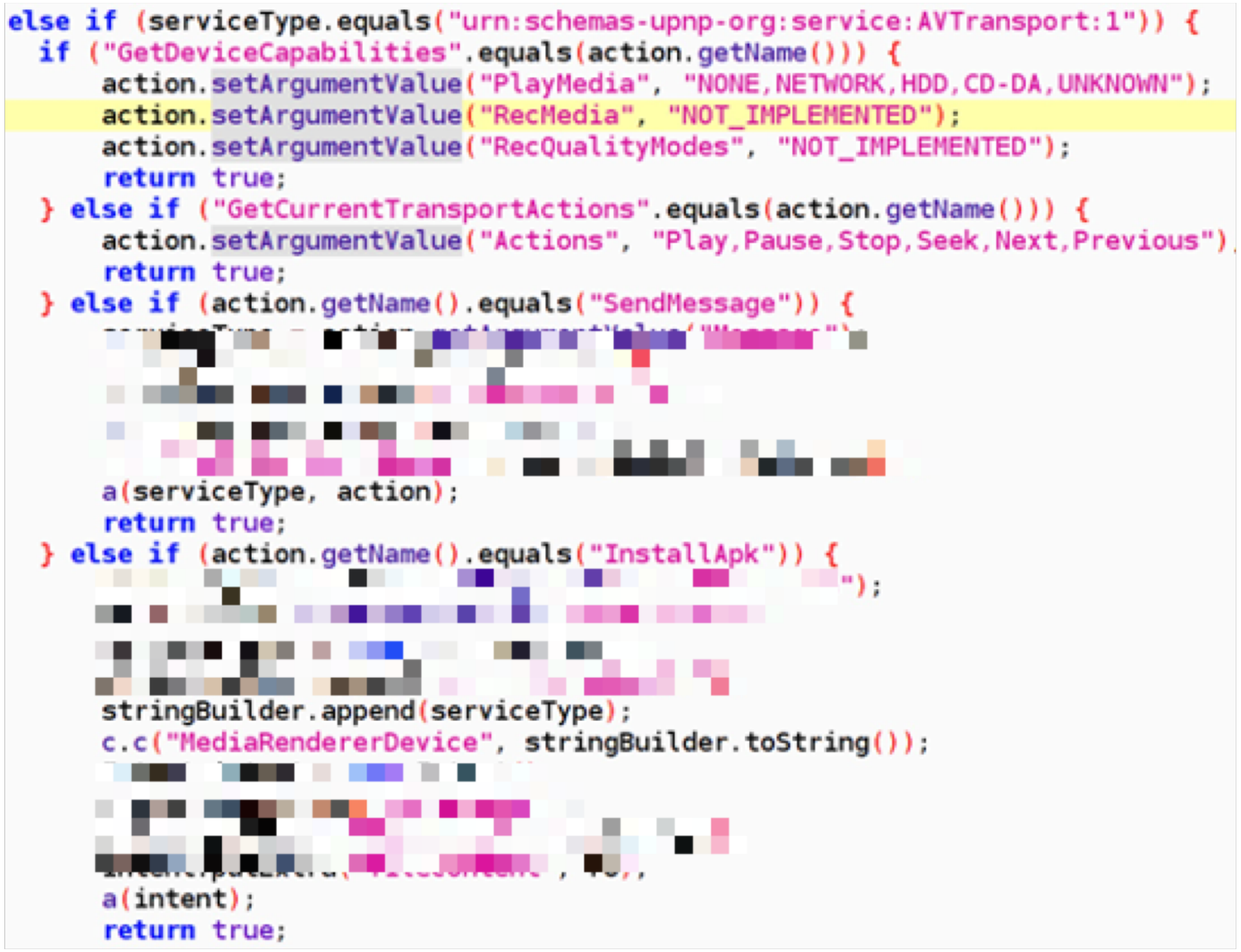
Attack Smart Tv Case 1



```
<serviceList>
    <service>
        <serviceType>urn:schemas-upnp-org:service:AVTransport:1</serviceType>
        <serviceId>urn:upnp-org:serviceId:AVTransport</serviceId>
        <SCPDURL>AVTransport.scpd.xml</SCPDURL>
        <controlURL>_urn:schemas-upnp-org:service:AVTransport_control</controlURL>
        <eventSubURL>_urn:schemas-upnp-org:service:AVTransport_event</eventSubURL>
    </service>
    <service>
        <serviceType>urn:schemas-upnp-org:service:ConnectionManager:1</serviceType>
        <serviceId>urn:upnp-org:serviceId:ConnectionManager</serviceId>
        <SCPDURL>ConnectionManager.scpd.xml</SCPDURL>
        <controlURL>_urn:schemas-upnp-org:service:ConnectionManager_control</controlURL>
        <eventSubURL>_urn:schemas-upnp-org:service:ConnectionManager_event</eventSubURL>
    </service>
    <service>
        <serviceType>urn:schemas-upnp-org:service:RenderingControl:1</serviceType>
        <serviceId>urn:upnp-org:serviceId:RenderingControl</serviceId>
        <SCPDURL>RenderingControl.scpd.xml</SCPDURL>
        <controlURL>_urn:schemas-upnp-org:service:RenderingControl_control</controlURL>
        <eventSubURL>_urn:schemas-upnp-org:service:RenderingControl_event</eventSubURL>
    </service>
</serviceList>
```

- Expose some Interface with no authorization
- Basically DLNA screen-mirroring
- Inject backdoor into Tv

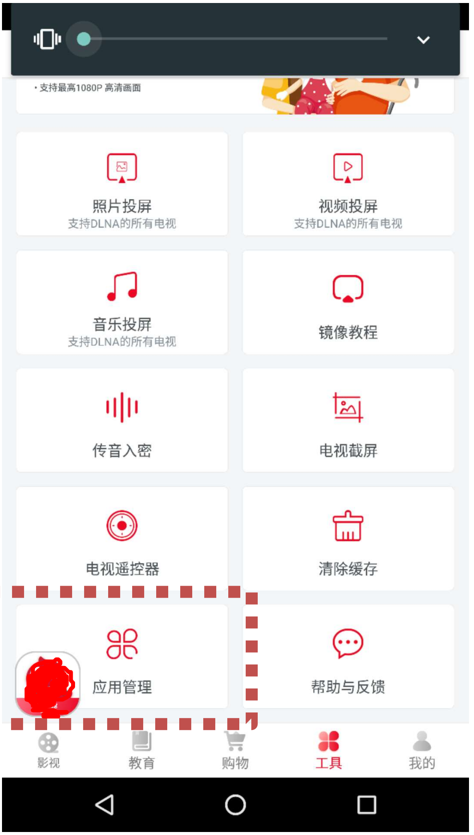
Attack Smart Tv Case 1

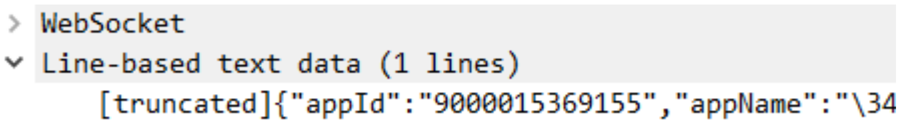


```
else if (serviceType.equals("urn:schemas-upnp-org:service:AVTransport:1")) {
  if ("GetDeviceCapabilities".equals(action.getName())) {
      action.setArgumentValue("PlayMedia", "NONE,NETWORK,HDD,CD-DA,UNKNOWN");
      action.setArgumentValue("RecMedia", "NOT_IMPLEMENTED");
      action.setArgumentValue("RecQualityModes", "NOT_IMPLEMENTED");
      return true;
 } else if ("GetCurrentTransportActions".equals(action.getName())) {
      action.setArgumentValue("Actions", "Play,Pause,Stop,Seek,Next,Previous")
      return true;
 } else if (action.getName().equals("SendMessage")) {
      [redacted]
      a(serviceType, action);
      return true;
 } else if (action.getName().equals("InstallApk")) {
      [redacted]
      stringBuilder.append(serviceType);
      c.c("MediaRendererDevice", stringBuilder.toString());
      [redacted]
      a(intent);
      return true;
```

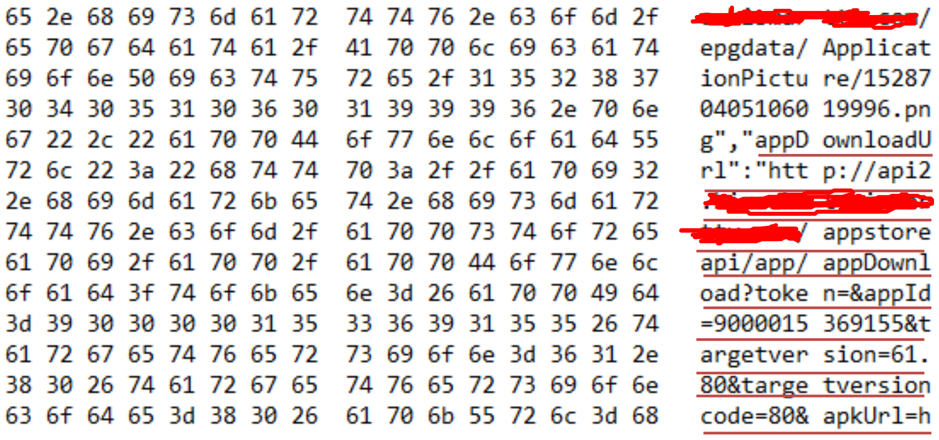
- Dangerous Upnp Action
- Remote Download->Install App->Launch App
- Attacker hjacked private network

Attack Smart Tv Case 2





```
> WebSocket
v Line-based text data (1 lines)
      [truncated]{"appId":"9000015369155","appName":"\34
```



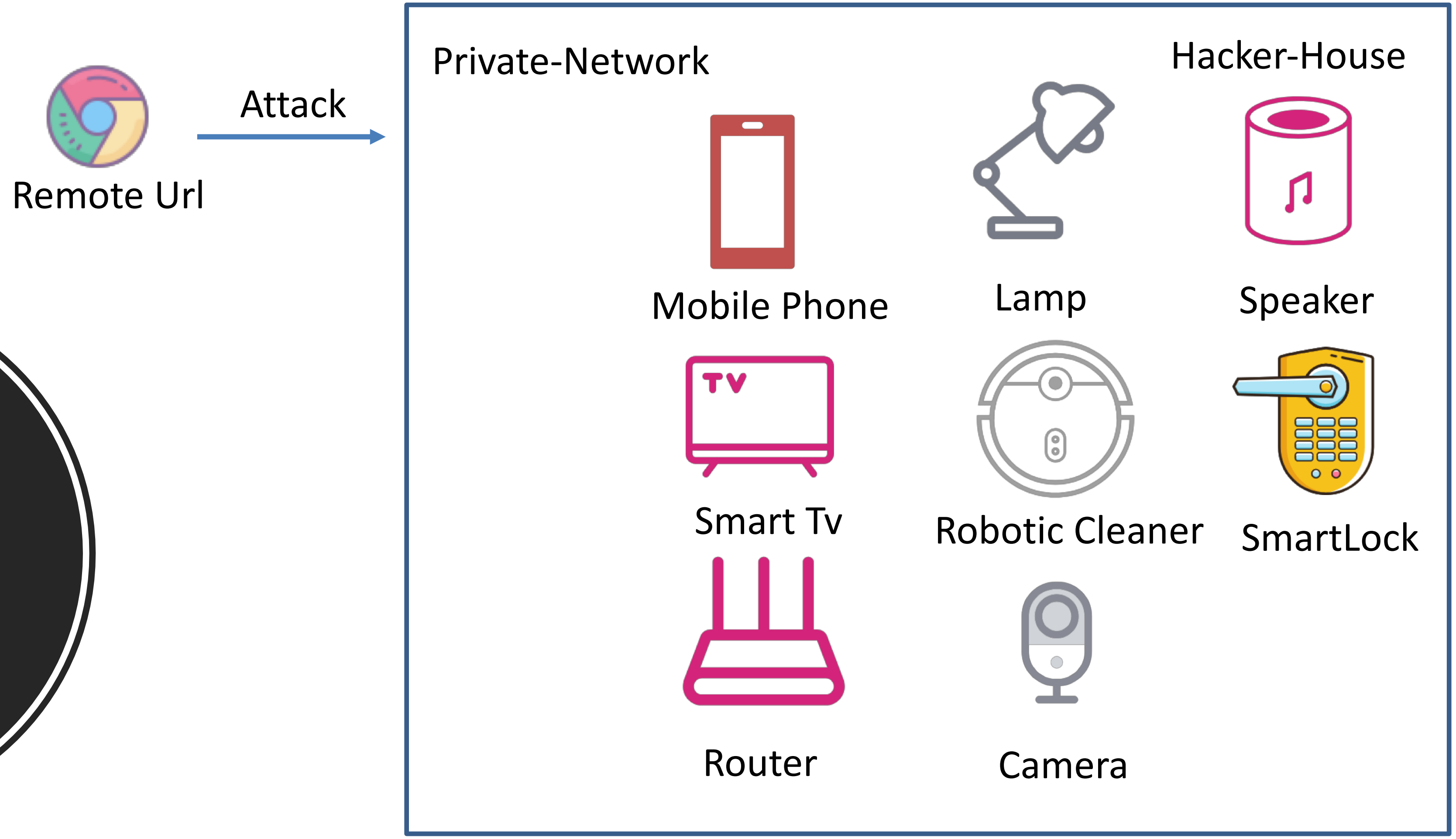
```
65 2e 68 69 73 6d 61 72  74 74 76 2e 63 6f 6d 2f   [redacted]/
65 70 67 64 61 74 61 2f  41 70 70 6c 69 63 61 74   epgdata/ Applicat
69 6f 6e 50 69 63 74 75  72 65 2f 31 35 32 38 37   ionPictu re/15287
30 34 30 35 31 30 36 30  31 39 39 39 36 2e 70 6e   04051060 19996.pn
67 22 2c 22 61 70 70 44  6f 77 6e 6c 6f 61 64 55   g","appD ownloadU
72 6c 22 3a 22 68 74 74  70 3a 2f 2f 61 70 69 32   rl":"htt p://api2
2e 68 69 6d 61 72 6b 65  74 2e 68 69 73 6d 61 72   [redacted]
74 74 76 2e 63 6f 6d 2f  61 70 70 73 74 6f 72 65   [redacted]/ appstore
61 70 69 2f 61 70 70 2f  61 70 70 44 6f 77 6e 6c   api/app/ appDownl
6f 61 64 3f 74 6f 6b 65  6e 3d 26 61 70 70 49 64   oad?toke n=&appId
3d 39 30 30 30 30 31 35  33 36 39 31 35 35 26 74   =9000015 369155&t
61 72 67 65 74 76 65 72  73 69 6f 6e 3d 36 31 2e   argetver sion=61.
38 30 26 74 61 72 67 65  74 76 65 72 73 69 6f 6e   80&targe tversion
63 6f 64 65 3d 38 30 26  61 70 6b 55 72 6c 3d 68   code=80& apkUrl=h
```

- ➢Weak App Code Protection
- ➢Communicate with Tv with no authorization
- ➢Remote attack Smart Tv imitate Center App Action

GeekPwn hacker-house case

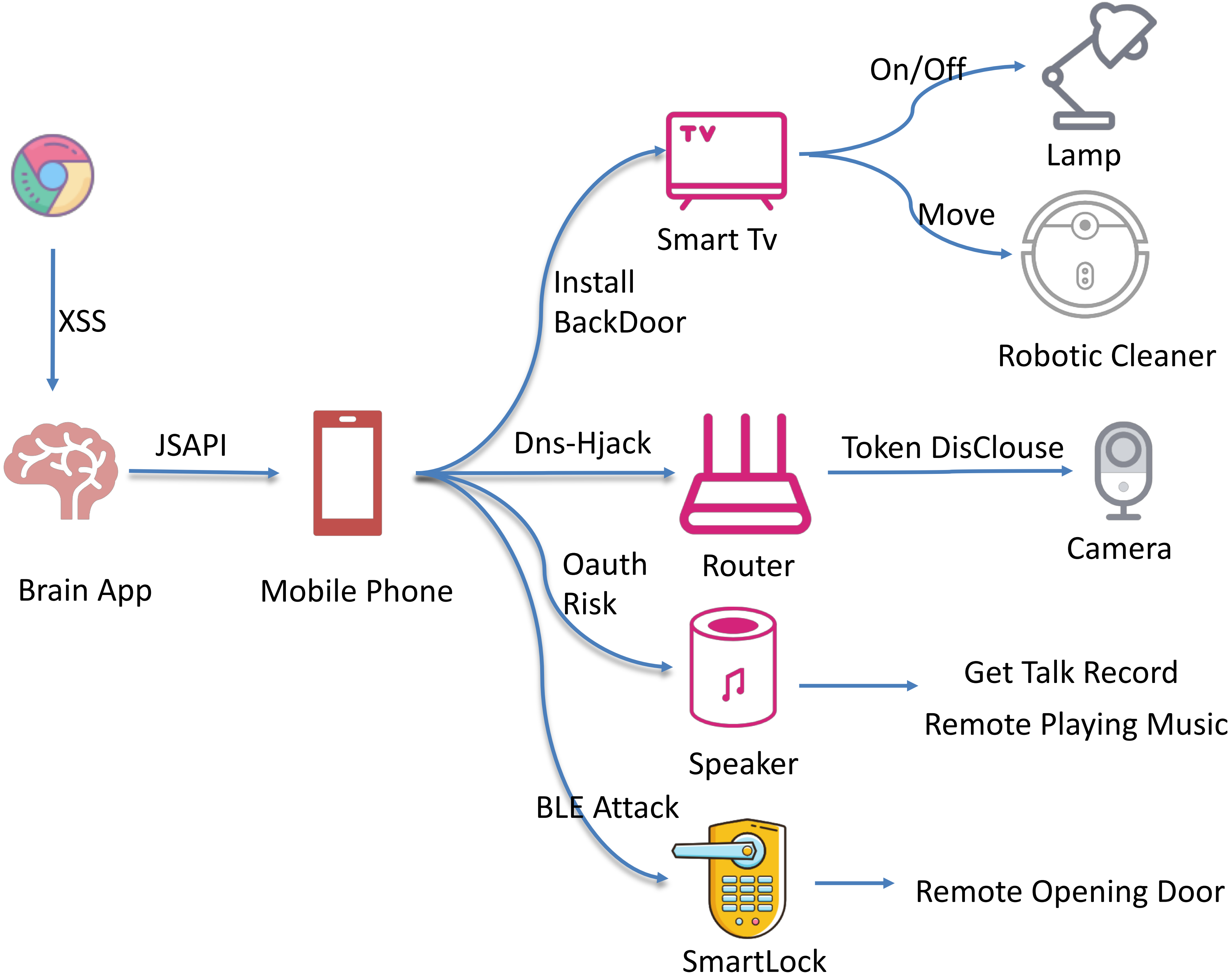
- A mini simulating smart home
- Pwn all of IoT devices in this virturl house
- Attack and hack IoT device from center brain app
- Expand and exploit JSAPI ability to access smart home control
- Achieve persistence and concealed

Hacker-House Case 1

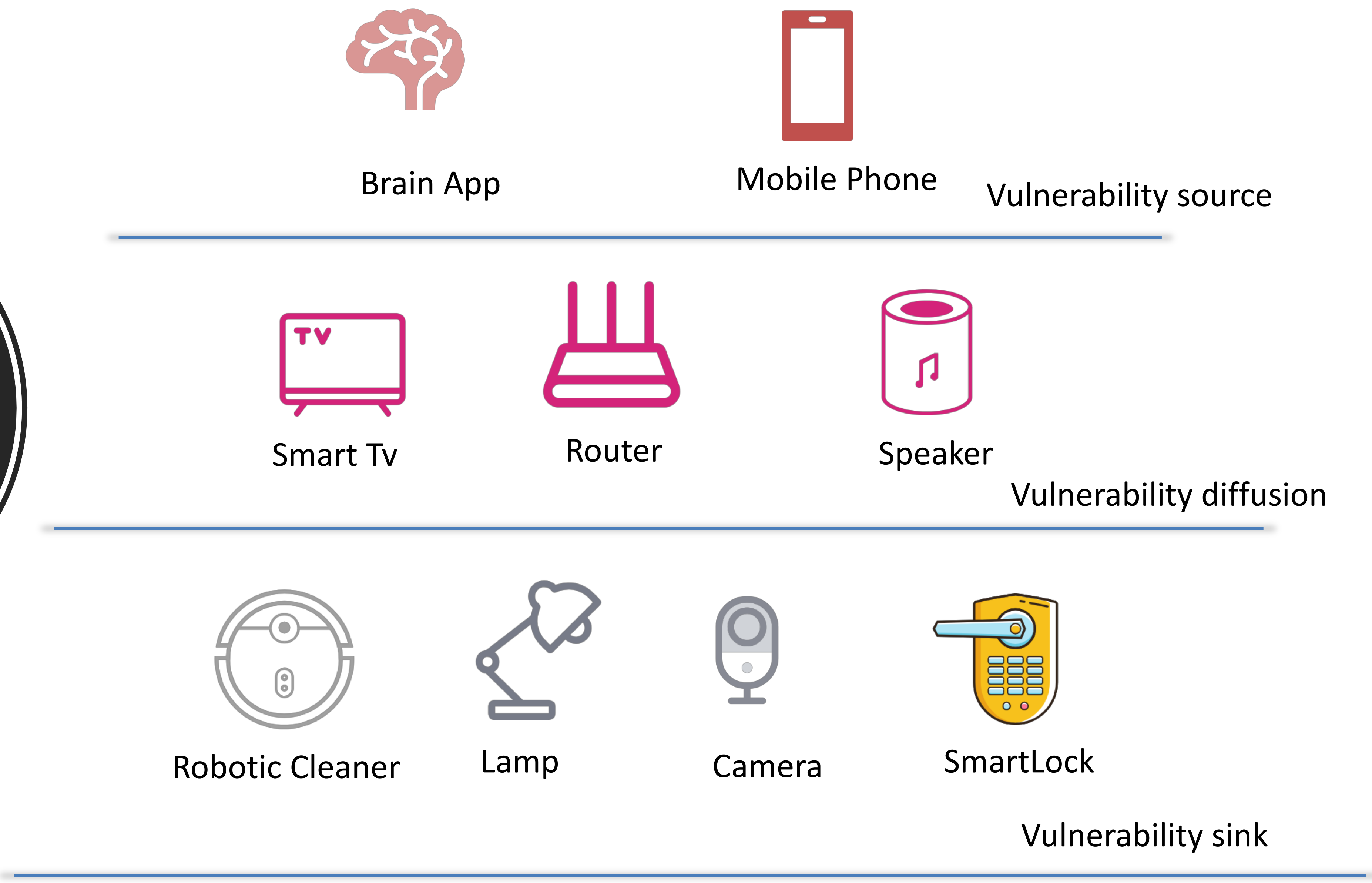


- Analyze and Choose Attack Surface
- Pwn target devices with obvious showing

Hacker-House Case 2



Hacker-House Case 3



Brain App

Mobile Phone

Vulnerability source

Smart Tv

Router

Speaker

Vulnerability diffusion

Robotic Cleaner

Lamp

Camera

SmartLock

Vulnerability sink

Conclusion

- We have found about 50 0-Day vulnerabilities in famous IoT Vendor within two month
  - Code Execution
  - Remote Contorl
  - Information Disclouse
  - Permanent denial of service
- We were ranked #1 in GeekPwn Hacker-House in 2018

Q&A

THANKs

Han Zidong@tencent