

SIP Hacking

Hendrik Scholz

<hscholz@raisdorf.net>

<http://www.wormulon.net/>

3rd Telephony Summit
Wiesbaden, Germany – 2006-05-02

Agenda

- VoIP overview
- specific Attacks
 - Forking/Traffic Amplification
 - End User Devices
 - Routing
 - Protocol Independent Attacks
 - Implementation Differences
 - Configuration Bugs

VoIP for Managers

- VoIP equals
 - cheap: run PSTN on Internet infrastructure
 - more features: ISDN + Instant Messaging
- in production use today
 - end users
 - peering/transit
- Google/Skype cannot be wrong
 - explosive growth

The Dark Side

- PSTN converges with the Internet
 - more old hardware to take care of
- PSTN features need to be implemented
 - fundamental differences

clever network + dumb terminals

goes

dumb network + clever applications

SIP Standards - Feel Lost?

1847, 2045, 2046, 2047, 2048, 2198, 2327, 2543, 2616, 2617, 2633,
2733, 2791, 2833, 2848, 2959, 2976, 3087, 3050, 3204, 3219, 3261,
3262, 3263, 3264, 3265, 3266, 3310, 3311, 3312, 3313, 3319, 3320,
3321, 3322, 3323, 3324, 3325, 3326, 3327, 3329, 3361, 3351, 3372,
3388, 3389, 3398, 3407, 3420, 3428, 3455, 3468, 3485, 3515, 3550,
3551, 3555, 3556, 3605, 3606, 3608, 3611, 3702, 3711, 3725, 3764,
3824, 3840, 3842, 3856, 3857, 3890, 3891, 3903, 3911, 3959, 3960,
3968, 3969, 3976, 4028, 4077, 4083, 4091, 4092, 4117, 4123, 4145,
4168, 4189, 4235, 4240, 4244, 4245, 4317, 4320, 4321, 4353, 4354,
4411, 4412

<http://www.packetizer.com/voip/sip/standards.html>

- 'few' additional drafts
- new RFCs/drafts on a weekly basis

Session Initiation Protocol

- Requests
 - i.e. INVITE, REGISTER, CANCEL
- Responses
 - i.e 200 OK, 403 Forbidden, 404 Not Found
- lots of additions
 - Caller ID (Remote Party ID, RFC 3323, RFC 3325)
 - supplementary services (HOLD, MCID, CCBS)
- complex state engine

Attack Vectors

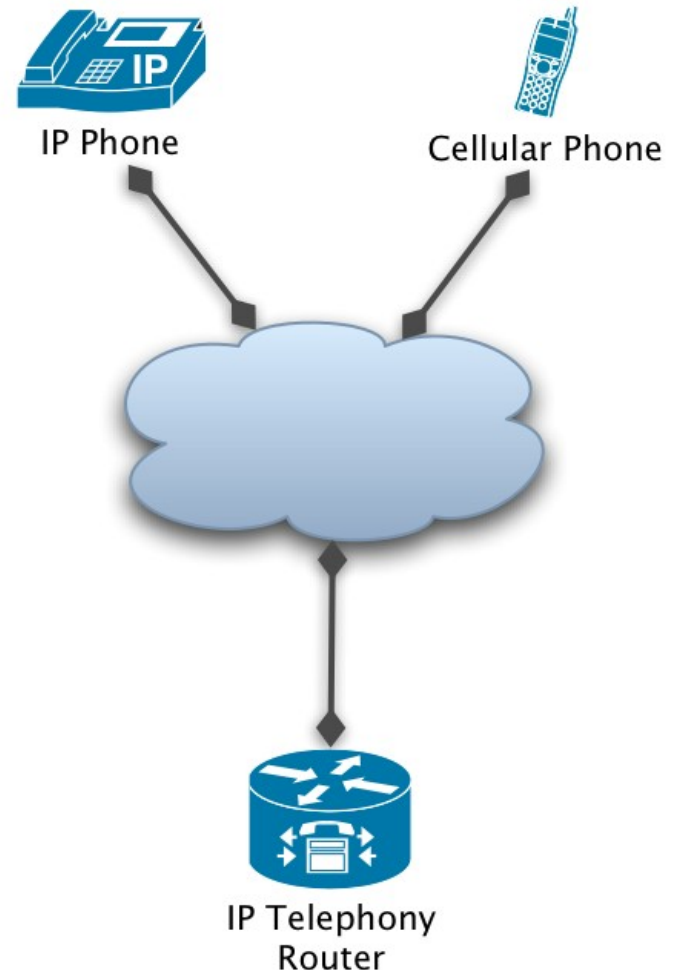
- Signalling (SIP)
- SIP Neighborhood: Billing, PSTN, MGCP, ...
- Routing
- End Devices
- Protocol independent attacks
- Implementation specific issues
- Configuration bugs

SIP Signalling

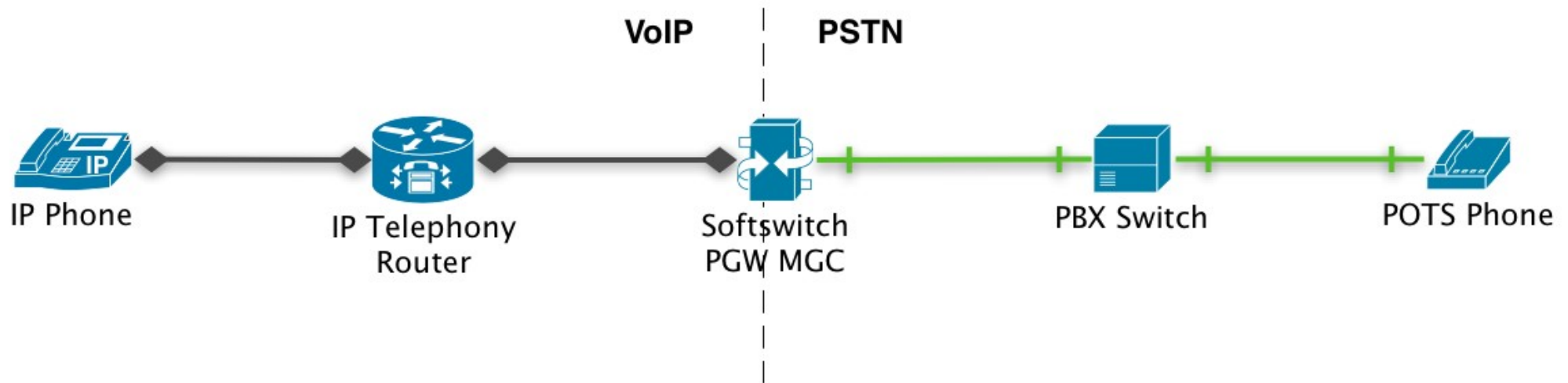
Singalling: Call Forking

- Call Forking
 - parallel/serial forking
 - wanted behaviour
- possible problems
 - traffic amplification
 - resource starvation (if stateful)

User	Contact
adam	adam@10.1.1.1:5060;tag=value
john	john@172.30.1.1:5060;opaque=123
john	john@192.168.1.1:18123;foo=bar



Signalling: Call Forwarding



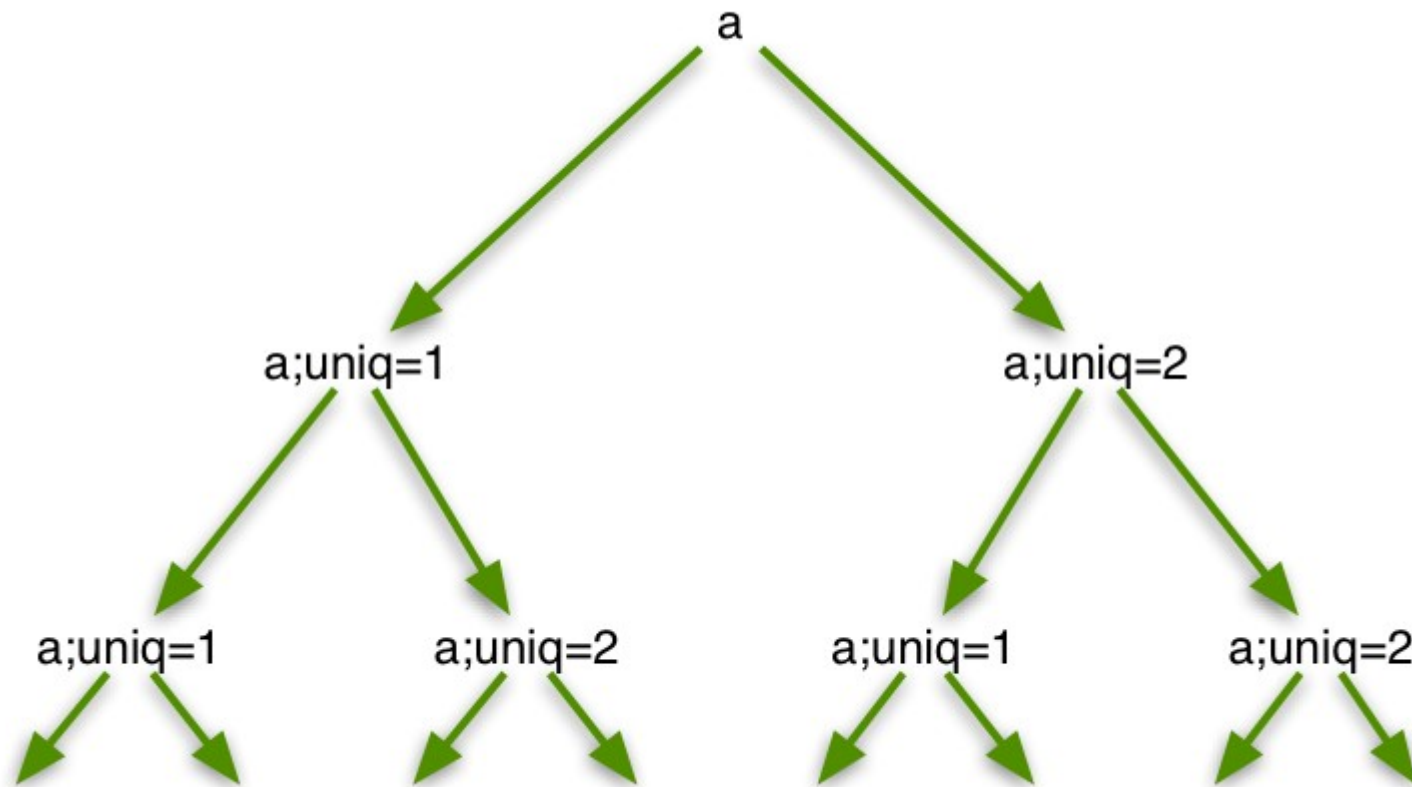
- Time To Live:
 - SIP: Max-Forwards (counts down)
 - SS7/ISUP: Redirect counter (counts up)
- Loops? they do happen

Fork Loop: Ingredients

- parallel call forking
 - two contacts for one user
- add the loop
 - strip IP from contact and add local domain
 - add tag to keep contact fields different

User	Contact
a	a@wormulon.net;uniq=1
a	a@wormulon.net;uniq=2

Fork Loop: Tree



source: draft-ietf-sip-fork-loop-00

Fork Loop: Preparation

- REGISTER users
 - <http://sipp.sf.net/>
 - <http://sipsak.org/>
- single call to user A
 - use your phone ;)
- wait for 2^{70} INVITEs to be processed
 - 1,180,591,620,717,411,303,424 INVITEs
 - 408 timeout will be triggered -> attack teared down

Fork Loop: Denial of Service

- add 3rd contact: victim
 - remote IP
 - random port
 - UDP or TCP transport
- network might die before victim does

Fork loop: Improvements

- PSTN contact
 - forward to cell phone
 - cell phone forwards back to SIP proxy
 - results in
 - new calls, fresh timeout, full TTL (Max-Forwards)
- Announcement contact
 - announcement starts playing immediately
 - redirect RTP/media to victim using SDP
- modify PSTN/announcement/fork ratio for more destruction

Routing

Routing Attacks

- Routing
 - based on dialplan inside each device, or
 - predefined Routes (Route/Service-Route header)
- use Route headers to direct messages
- REGISTER at foreign site

End User Devices

End User Devices

- routers/modems/PBXs/ATAs
 - Operating System
 - Unpatched
 - Unprotected
 - No logging/notification
 - Web interface
 - ISP-wide monocultures

End User Devices: Attacks

- little CPU power, limited number of lines
 - resource starvation
- no inbound Authentication
 - needed for ENUM et al.
 - SPIT
- remote management
 - reboot, config change, call control, click to dial

Locating Devices

- smap
 - mashup of sipsak and nmap
- available at <http://www.wormulon.net/>
- utilize OPTIONS SIP request
- basic banner grabbing for fingerprinting
- 80-90% VoIP enabled devices observed!

Locating Devices: smap output

```
$ smap -O -t 200 89.53.10.0/24

scanning 89.53.10.0... timeout
scanning 89.53.10.1... timeout
....
scanning 89.53.10.8... up
User-Agent: AVM FRITZ!Box Fon WLAN 7050 14.04.01 (Jan 25 2006)
scanning 89.53.10.9... up
User-Agent: AVM FRITZ!Box Fon WLAN 7050 14.04.01 (Jan 25 2006)
scanning 89.53.10.10... up
User-Agent: AVM FRITZ!Box Fon WLAN 7050 14.04.01 (Jan 25 2006)
...

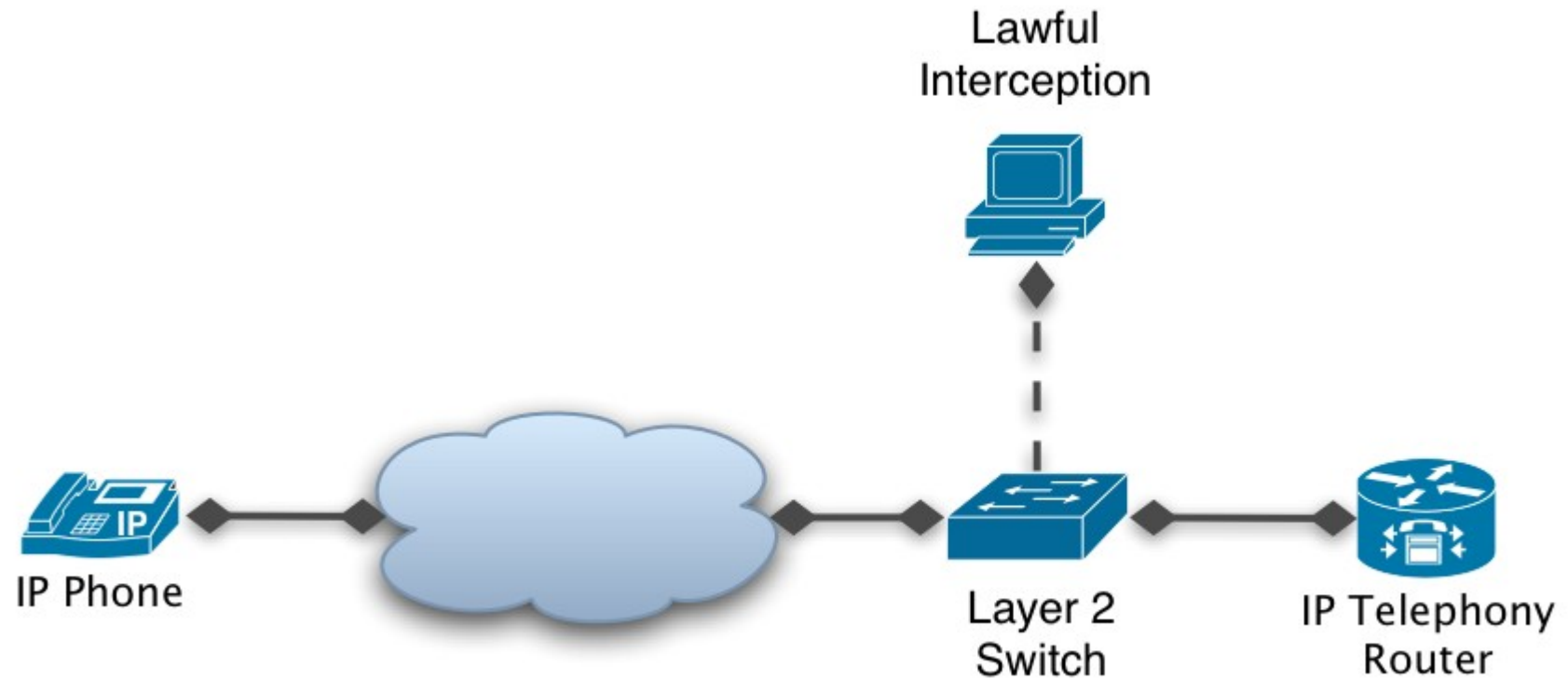
256 hosts scanned, 114 up, 142 down, 0 errors
$ nmap -sP 89.53.10.0/24
...
Nmap run completed -- 256 IP addresses (138 hosts up) scanned in
5.400 seconds
$
```

Protocol Independent Attacks

Timing Attacks

- exploit UDP defragmentation timer
- evade billing & Lawful Interception
- inspired by Van Hauser's IPv6 talk at 22C3
- goal: fool passive Lawful Interception

Timing Attack: LI Setup



Timing Attack: LI Box

- receives mirrored traffic
- use libnids defragmentation in userland
- parse SIP messages (i.e. using libosip)
- check username/phone # against DB
- copy message to LEA if needed

Timing Attack: Timers

- two different IP stacks
 - different implementation
 - different configuration
- LI Box might drop fragments too early
- ... or too late
- goal: prevent a messages from being de-fragmented on LI system

LI timer < LIVE timer

- inject 1st fragment
 - LI stores fragment
 - LIVE stores fragment
- wait for fragment to expire on LI system
- inject 2nd fragment
 - LIVE de-fragments successfully
 - LI system stores second fragment

LI timer > LIVE timer

- inject 1st fragment: fill both buffers
- wait for LIVE system to drop fragment
- inject 2nd fragment
 - LI box de-fragments successfully
 - LIVE stores fragment
- inject 3rd fragment
 - LI box stores fragment
 - LIVE de-fragments and initiates call

SIP Implementation Differences

RFC 3261 Implementation

- RFC 3261 To/From 'Displayname'
- Displayname considered a comment
- libosip bails out on comma in Displayname
- osip_message_parse() fails

- add comma in Displayname and break LI system previously described

Implementation: Caller-ID

- Different implementations
 - From
 - Remote-Party-ID
 - P-Preferred-Identity/P-Asserted-Identity
 - ISP proprietary extensions, i.e. SetCallerID:

Implementation: Caller-ID

- spoof Caller-ID using different implementation
- set to cell phone number, call voice mail

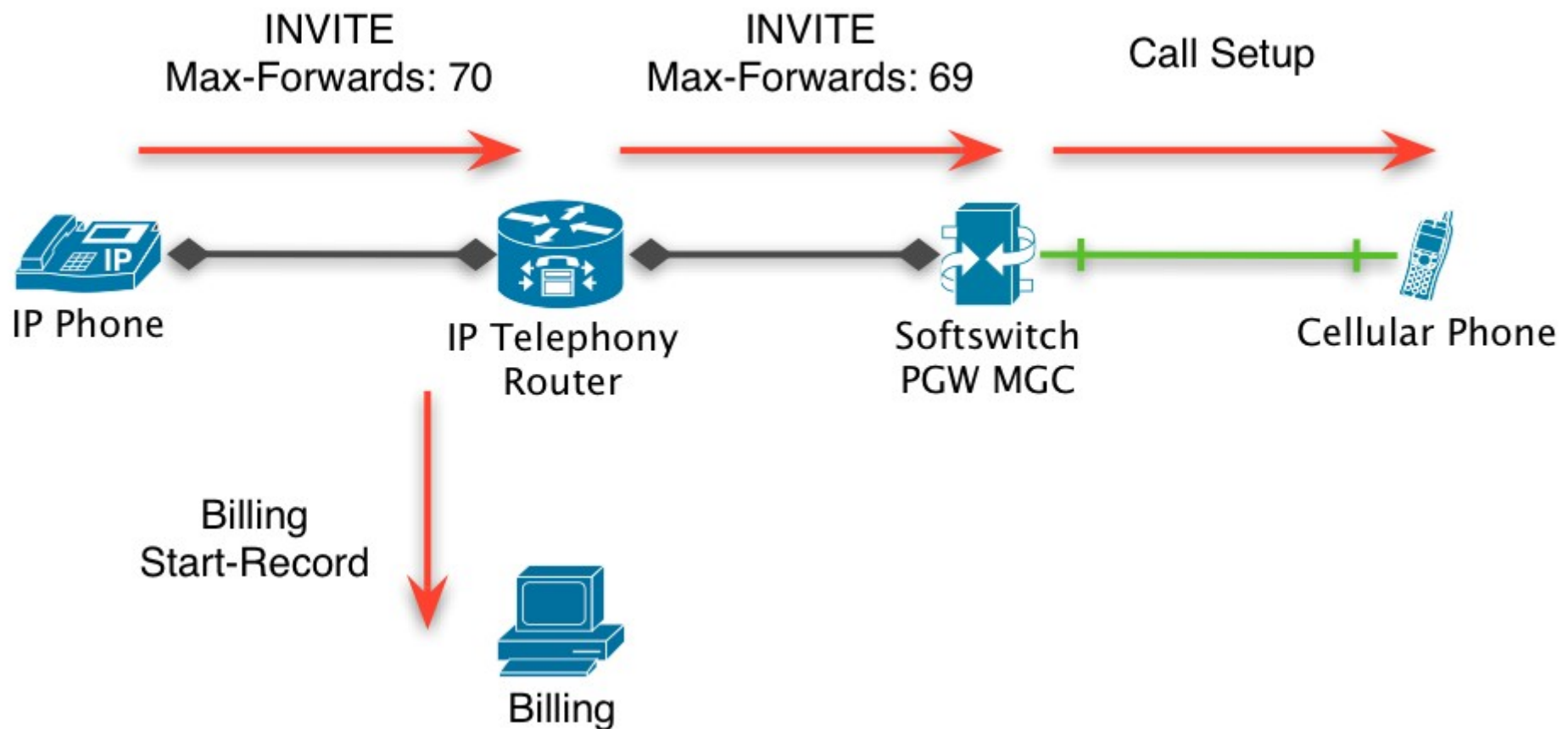


```
INVITE sip:0049311@123.org
From: "foo" <0049199123@123.org>
Remote-Party-ID: <sip:001800999@123.org>
P-Asserted-Identity: <sip:001800999@123.org>
Authorization: ... username="foo" ...
```

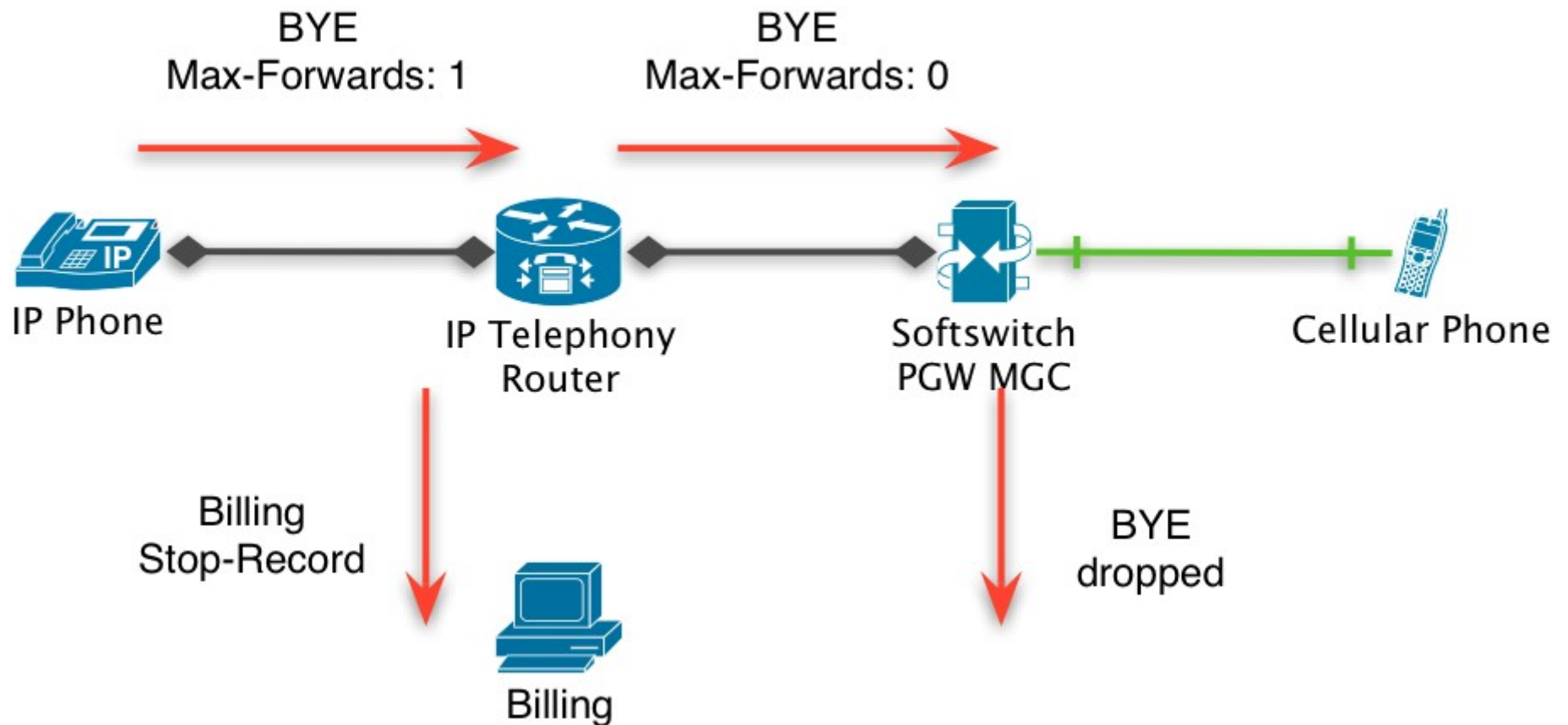
```
INVITE sip:0049311@123.org
From: "foo" <0049199123@123.org>
Remote-Party-ID: <sip:0049199123@123.org>
P-Asserted-Identity: <sip:001800999@123.org>
```

Configuration Bugs

Max-Forwards: cheap calls



Max-Forwards: cheap calls



Conclusions

- PSTN Convergence still in progress
 - new hardware
 - new RFCs (ISDN supplementary services)
 - regional laws (LNP, emergency calls, ...)
- Research areas
 - fingerprinting, stack differences
 - SPIT
- Filesharing

Upcoming Events

- 3rd VoIP Security Workshop
 - Berlin, Germany
- Syscan '06
 - Singapore
- Bellua Cyber Security '06
 - Jakarta, Indonesia
- Sipit 19
 - Durham, NH, USA

Resources

- Call Cases: <http://www.tech-invite.com/>
- Documentation: <http://www.softarmor.com/>
- SIP software
 - SER: <http://iptel.org/>
 - OpenSER: <http://openser.org/>
 - Asterisk: <http://asterisk.org/>
 - sipp, sipsak
 - Protos Test Suite

Questions & Answers

Q&A

hscholz@raisdorf.net

