

# Attacking VoIP Networks

Hendrik Scholz  
Freenet Cityline GmbH

<hendrik.scholz@freenet-ag.de>

**freenet.de**  
normal ist das nicht!

# <http://Freenet.de/>

- German ISP + PSTN carrier
- > 700,000 DSL customers
- high VoIP acceptance
- VoIP enabled routers (AVM Fritz!, Siemens, ...)
- VoIP used as PSTN replacement

# Attack Vectors

# Attack Vectors

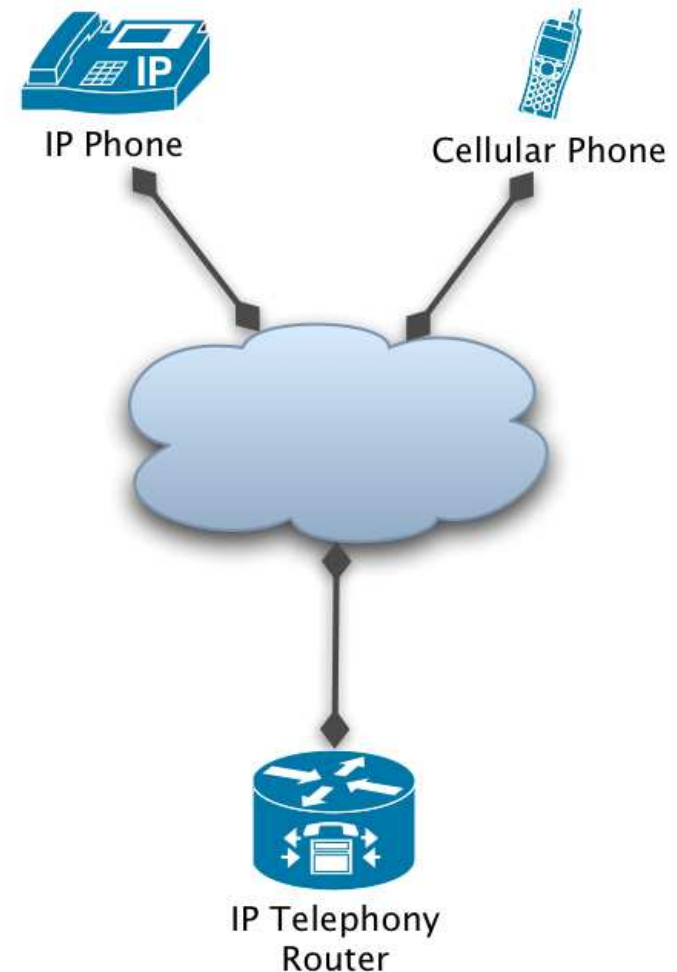
- focus on
  - Server (D)DOS: Traffic Amplification
  - End Devices (CPE)
  - Unintentional DDOS
  - Protocol Independent Issues
- more
  - Billing Evasion
  - Identity Theft, Privacy (german Laws apply)
  - ...

# Amplification Attack

# Call Forking

- Call Forking
  - parallel/serial forking
  - wanted behaviour
- possible problems
  - traffic amplification
  - resource utilization

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

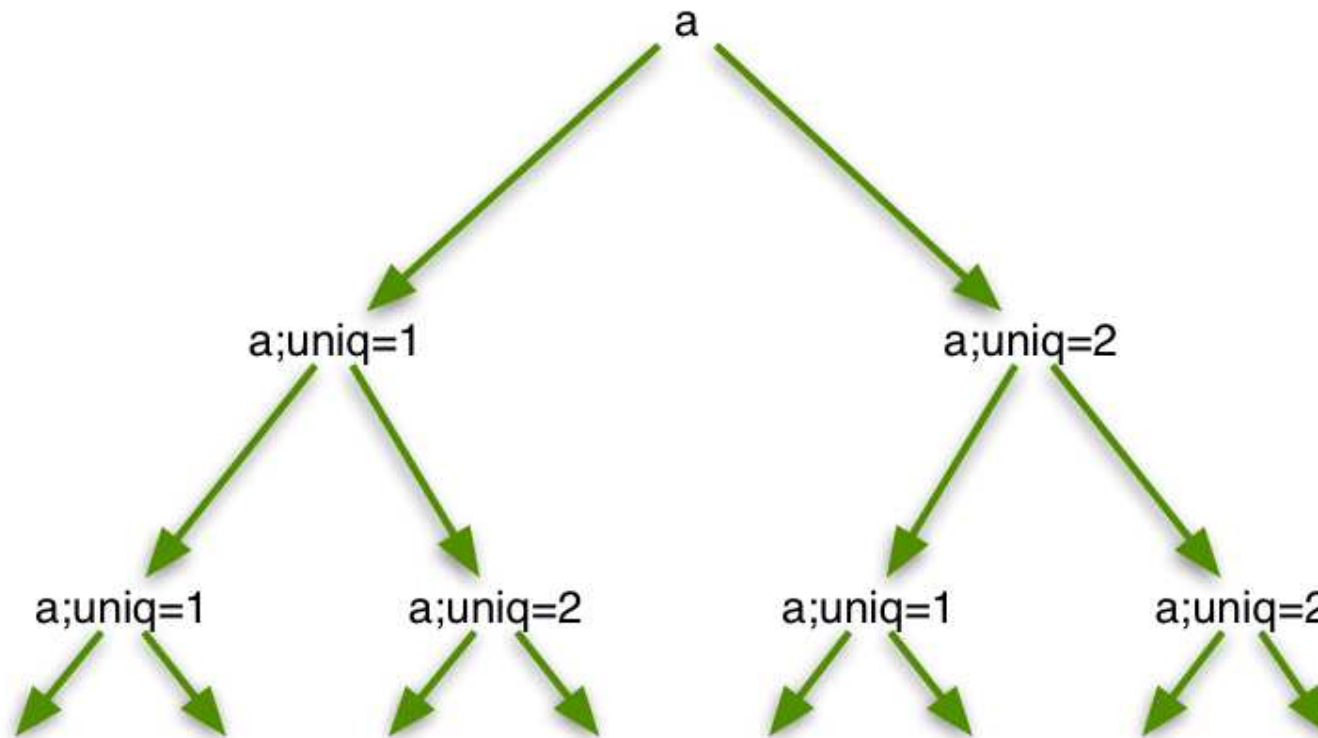


# Fork Loop Outline

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

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

# Fork Loop Tree



see also: [draft-ietf-sip-fork-loop-00](#)

# Fork Loop Preparation

- REGISTER contacts
  - use sipp or sipsak
- inject call to user A
  - use a phone
- wait for  $2^{70}$  INVITEs to be processed
  - 1180591620717411303424 INVITEs
  - 408 timeout will fire and tear down attack

# Fork Loop Enhancements

- TCP contact
  - SYN flood a random website
- PSTN contact
  - forward to cell phone and back to SIP proxy
  - results in new calls, fresh timers, full TTL
- Announcement contact
  - starts playing immediately (183 Session Progress)
  - redirect RTP to random victim using SDP
- modify PSTN/announcement/fork ratio

# End User Devices

# About End User Devices

- CPEs/PBXs/ATAs/...
  - Operating System (Linux, VxWorks)
  - Unpatched
  - No logging/notification
  - Web interface
  - ISP-wide monocultures

# Locating Devices

- smap
  - mashup of sipsak and nmap
  - available at <http://www.wormulon.net/>
  - utilize SIP OPTIONS request
  - basic banner grabbing for fingerprinting
- scan DSL ISP 'dial up' ranges
- up to 90% hit ratio
- common: 60-70% VoIP enabled

# smmap output

```
$ smmap -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
$
```

# CPE Attacks

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

# Unintentional DDOS

# What's that supposed to be?

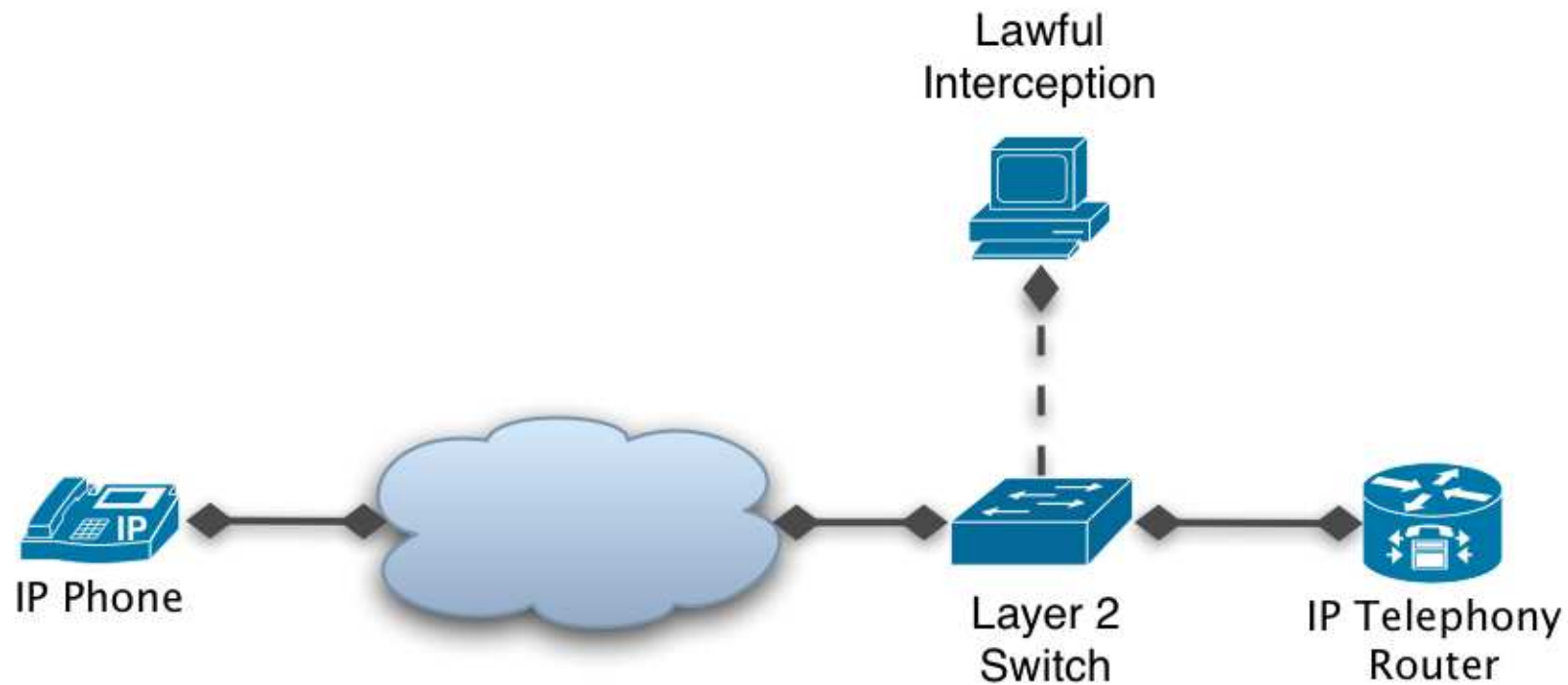
- AVM Fritz!Box series routers/ATAs
  - multiple generations
  - same firmware train
- new feature: **'DSL reconnect between 3 and 4AM'**
  - 'fetch bindings'
  - de-register own contact
  - DSL de-/reconnect
  - REGISTER
  - SUBSCRIBE for MWI

# Protocol Independent Attack

# Timing Attack

- exploit UDP defragmentation timer
- evade Billing & Lawful Interception
- inspired by Marc Heuse's IPv6 talk at 22C3
  
- goal: fool passive Lawful Interception

# Lawful Interception Setup



# Lawful Interception System

- receives mirrored traffic
- use libnids defragmentation in userland
- parse SIP messages (i.e. using libosip2)
- check username/phone # against DB
- copy message to LEA if needed

# LI Timing Attack

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

# LI timer < Live timer

- inject 1<sup>st</sup> fragment
  - 'LI' stores fragment
  - 'Live' stores fragment
- wait for fragment to expire on 'LI'
- inject 2<sup>nd</sup> fragment
  - 'Live' de-fragments successfully
  - 'LI' stores fragment

# LI timer > Live timer

- inject 1<sup>st</sup> fragment
- wait for 'Live' to drop fragment
- inject 2<sup>nd</sup> fragment
  - 'LI' defragments successfully
  - 'Live' stores fragment
- inject 3<sup>rd</sup> fragment
  - 'LI' stores fragment
  - 'LIVE' defragments and initiates call

# Conclusions

- lots of issues
  - do not underestimate German Laws
  - pay attention to Privacy & Trust between Entities
- few things interesting really happen
  - account creation to get free hardware
  - sign up & use free PSTN minutes
- Research on Intrusion Detection/Prevention

# Questions?



**<hendrik.scholz@freenet-ag.de>**

**Btw: We're hiring!**