Analysing Protocol Implementations

Anders Moen Hagalisletto, Lars Strand,
Wolfgang Leister and Arne-Kristian Groven

The 5th Information Security Practice and Experience Conference (ISPEC 2009)
Xi'an, China
April 2009

Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Outline

Motivation

VoIP

SIP – what is it? How does it work?

Method and tools – analysing implementation rather than the specification

Results

Conclusion

EUX2010SEC project goal

"The overall goal of this research project is to improve both the security level and the security awareness when developing, installing and using open source VoIP/PBX/multimedia solutions."

Several industry partners in Norway participating

VoIP

- Voice over IP (VoIP) protocols and technology is a merge of telecom and data communication
- What is VoIP?
  - Broad definition: Sending and receiving media (voice/video) over IP
- Why VoIP?
  - Added functionality and flexibility – which may be hard to provide over PSTN
  - Reduced cost – uses Internet as carrier
  - Less administration – no separate telephone and data network
- Industry have high focus on VoIP today
- But, VoIP is known to be insecure
  - Inherits problems from traditional IP networks
  - Multiple attack on SIP based VoIP exists

SIP

- Session Initiation Protocol (SIP) is the *de facto* standard signaling protocol for VoIP
  - Application layer (TCP, UDP, SCTP)
  - Setting up, modifying and tearing down multimedia sessions
  - Not media transfer (voice/video)
  - Establishing and negotiating the *context* of a call
- RTP transfer the actual multimedia
- SIP specified in RFC 3261 published by IETF 2002
  - First iteration in 1999 (RFC2543) – ten years old
  - Additional functionality specified in over 120 different RFCs(!)
  - Even more pending drafts...
  - Known to be complex and sometimes vague – difficult for software engineers to implement
  - Interoperability conference - “SIPit”

Excerpts from an email posted on IEFT RAI mailing list:

I'm finally getting into SIP. I've got Speakeasy VoIP service, two sipphone accounts, a Cisco 7960 and a copy of x-ten on my Mac.

And I still can't make it work. Voice flows in one direction only. I'm not even behind a NAT or firewall -- both machines have global addresses, with no port translations or firewalls.

*I've been working with Internet protocols for over 20 years. I've implemented and contributed to them. And if *I* can't figure out how to make this stuff work, how is the average grandmother expected to do so?* ***SIP is unbelievably complex, with extraordinarily confusing terms.*** *There must be half a dozen different "names" -- Display Name, User Name, Authorization User Name, etc -- and a dozen "proxies". Even the word "domain" is overloaded a half dozen different ways. This is ridiculous!*

Sorry. I just had to get this off my chest. Regards,

Reference: http://www.ietf.org/mail-archive/web/rai/current/msg00082.html

SIP - Basic terminology

- User Agent Client (UAC)
  - Endpoint, initiate SIP transaction
- User Agent Server (UAS)
  - Handles incoming SIP requests

} User agent

- Redirect Server
  - Retrieves address for callee and returns them to caller
- Proxy (server)
  - Autonomously processes and routes requests
- Registrar
  - Stores explicitly registered user addresses
- Location server
  - Provides information about a target user's location

SIP main functions

- INVITE Initiates a call signaling sequence
- BYE Terminates a session
- ACK Acknowledge
- OPTION Queries a server about its capabilities.
- CANCEL Cancel a request in progress.
- REGISTER Register location information at a registrar server.

SIP message syntax - INVITE

Start line (method)

```
INVITE sip:bob@NR SIP/2.0
```

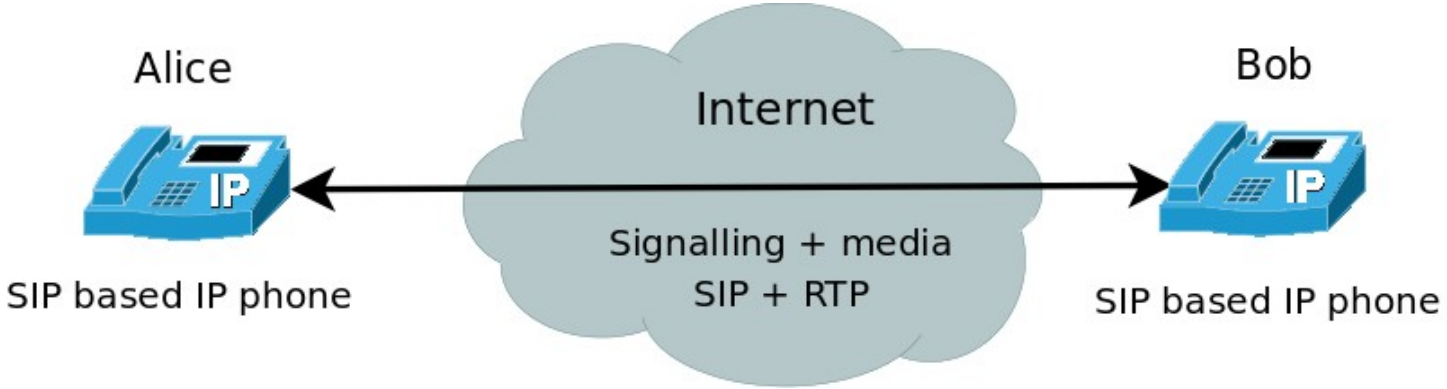
Message headers

```
Via: SIP/2.0/UDP 156.116.8.106:5060;rport;branch=z9hG4bK
From: Alice <sip:alice@NR>;tag=2093912507
To: <sip:bob@NR>
Contact: <sip:alice@156.116.8.106:5060>
Call-ID: 361D2F83-14D0-ABC6-0844-57A23F90C67E@156.116.8
CSeq: 41961 INVITE
Max-Forwards: 70
Content-Type: application/sdp
User-Agent: X-Lite release 1105d
Content-Length: 312
```

Message body (SDP content)

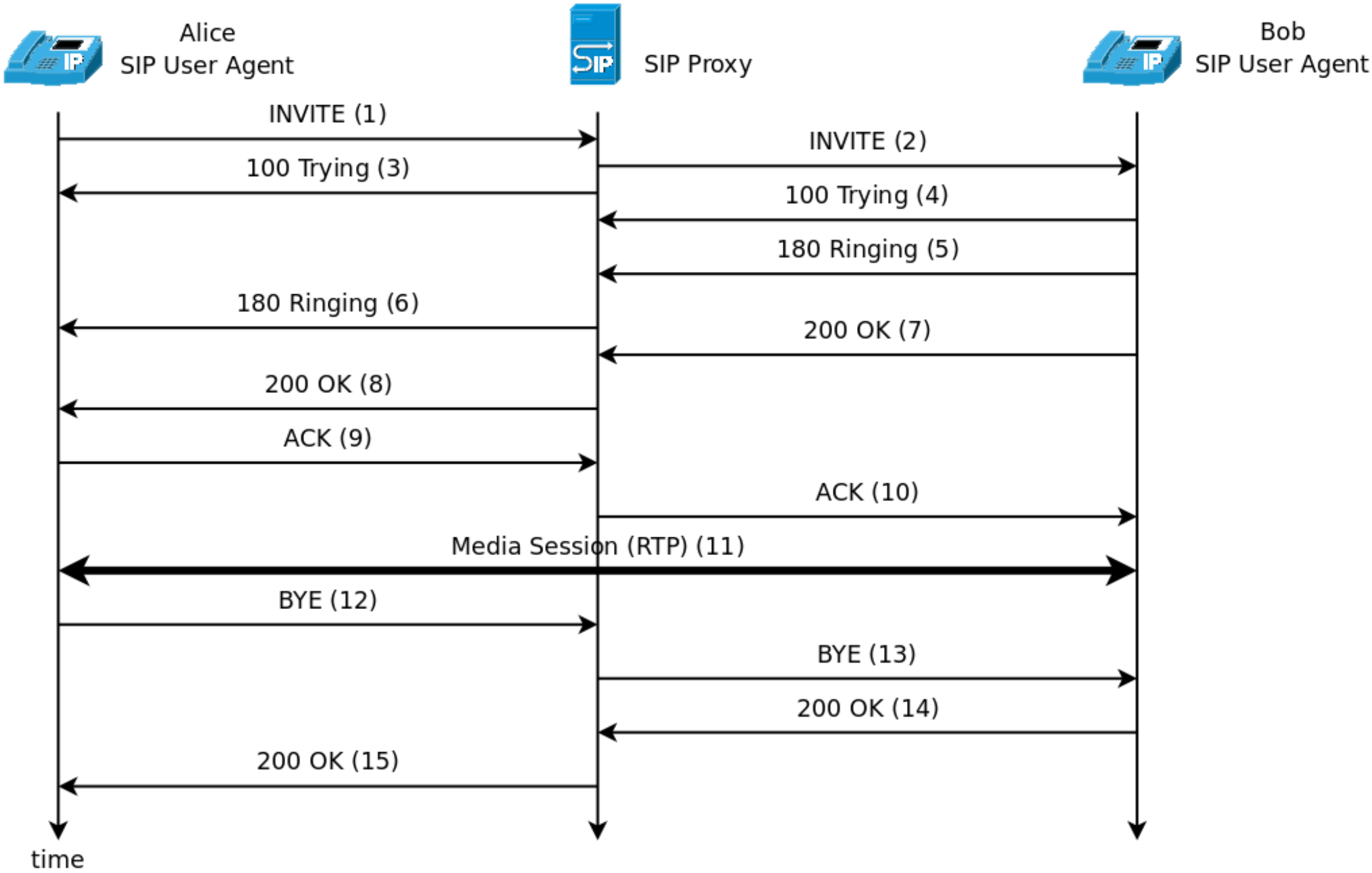
```
v=0
o=alice 2060633878 2060633920 IN IP4 156.116.8.106
s=SIP call
c=IN IP4 156.116.8.106
t=0 0
m=audio 8000 RTP/AVP 0 8 3 98 97 101
.............
```

SIP example
Direct call UA to UA



- Caller must know callee's IP or hostname
- No need for intermediate SIP hosts
- **Problems:**
  - Traversing firewalls
  - Seldom know IP/hostname of user
  - Mobility – change IP/hostname

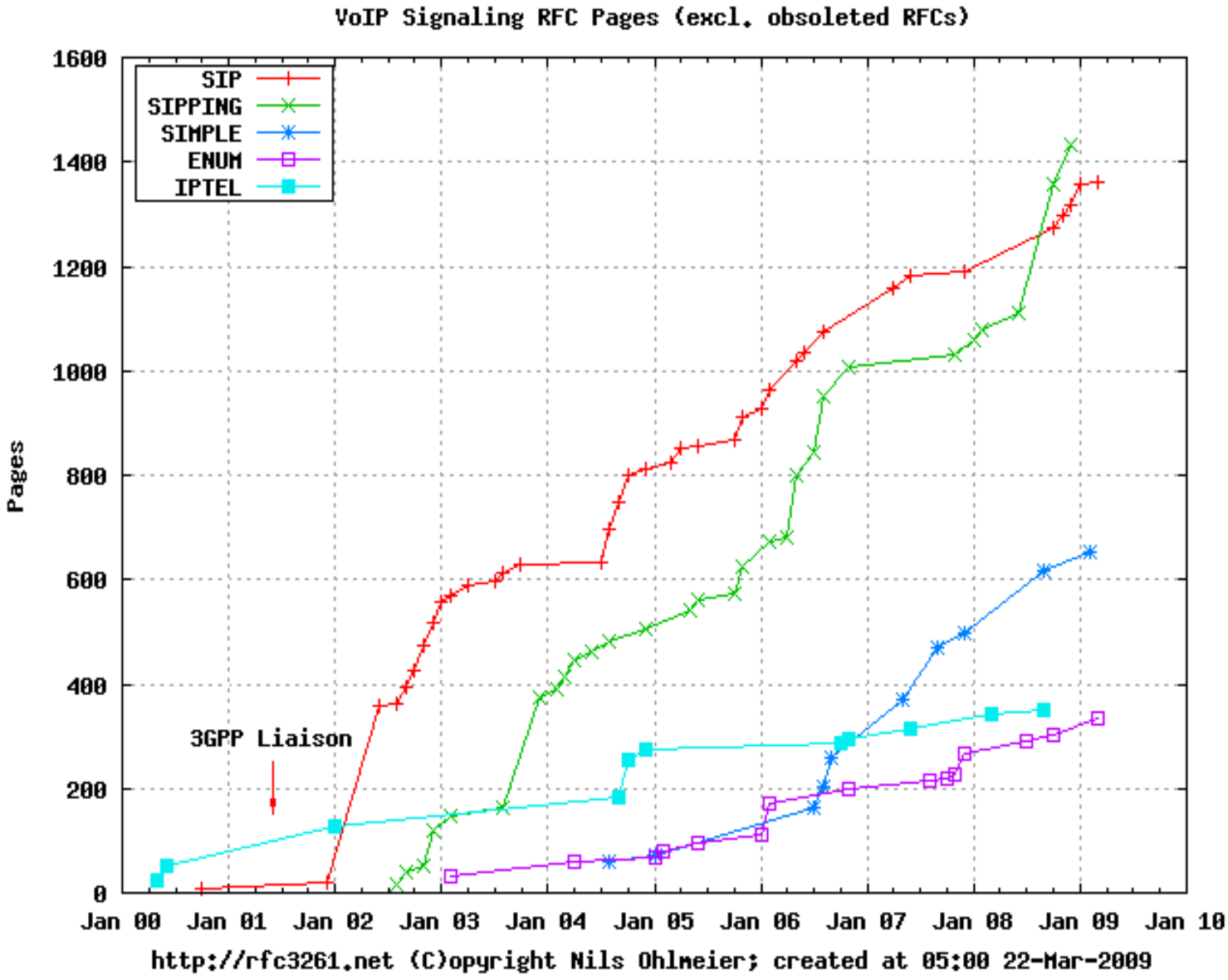
SIP example – proxied call



Method

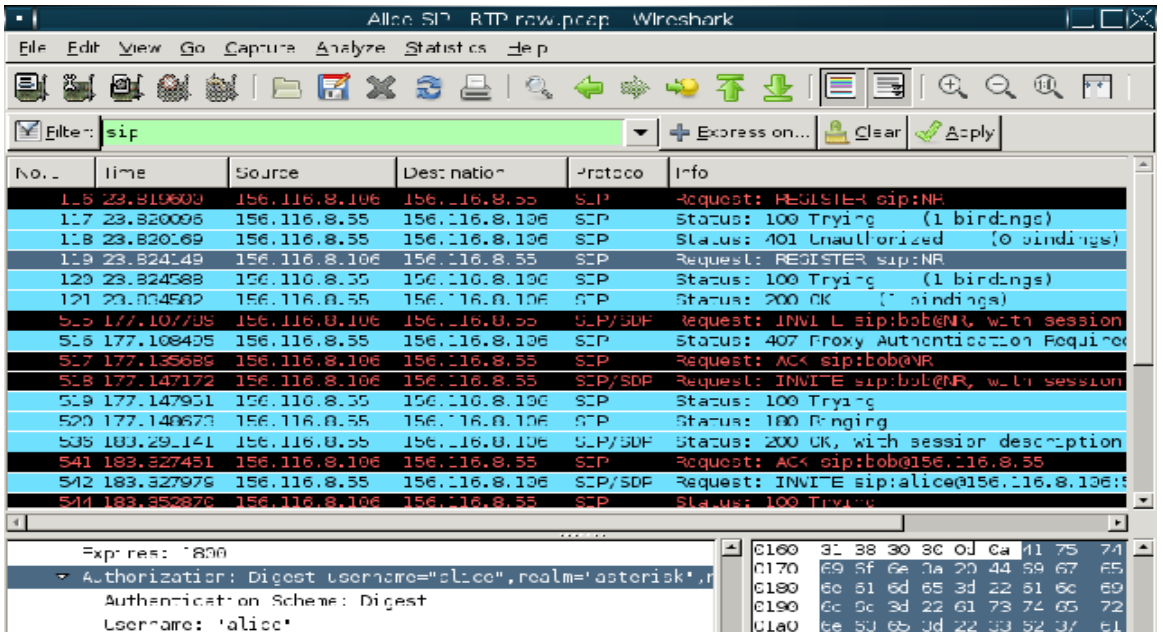
1) Experiment
   - *“Don't trust the documentation”*
   - Lab test setup: Set up a working VoIP environment
   - Software: Asterisk PBX and different soft/hardphones
2) Active observation
   - Capture network dumps using tcpdump
   - Analysed with “Wireshark”
3) Formal protocol analysis
   - PROSA

Read specification – huge, complex and sometimes vague



Network dump of VoIP traffic

- Using a network monitor and analysing tools
  - Tcpdump and Wireshark
- Why did we use Wireshark?
  - Learn and understand the standard
  - Compare implementations against the SIP standard
  - Results used as basis for modeling in PROSA
  - Formalization is done much faster when reading network dump than the standard alone



Why use formal methods?

Because

- the only way to prove or verify that protocols fulfills their goals!
- has been used to find new attacks on protocols
- Implicitly gives a unambiguous specification of
  - the protocol's interactions and entities
  - the functional and security goals
- the protocol specification can be analyzed automatically

Using the protocol analyser PROSA

- Static validation, written in Maude
- Developed by Anders Hagalisletto (PhD thesis)

Findings and implications

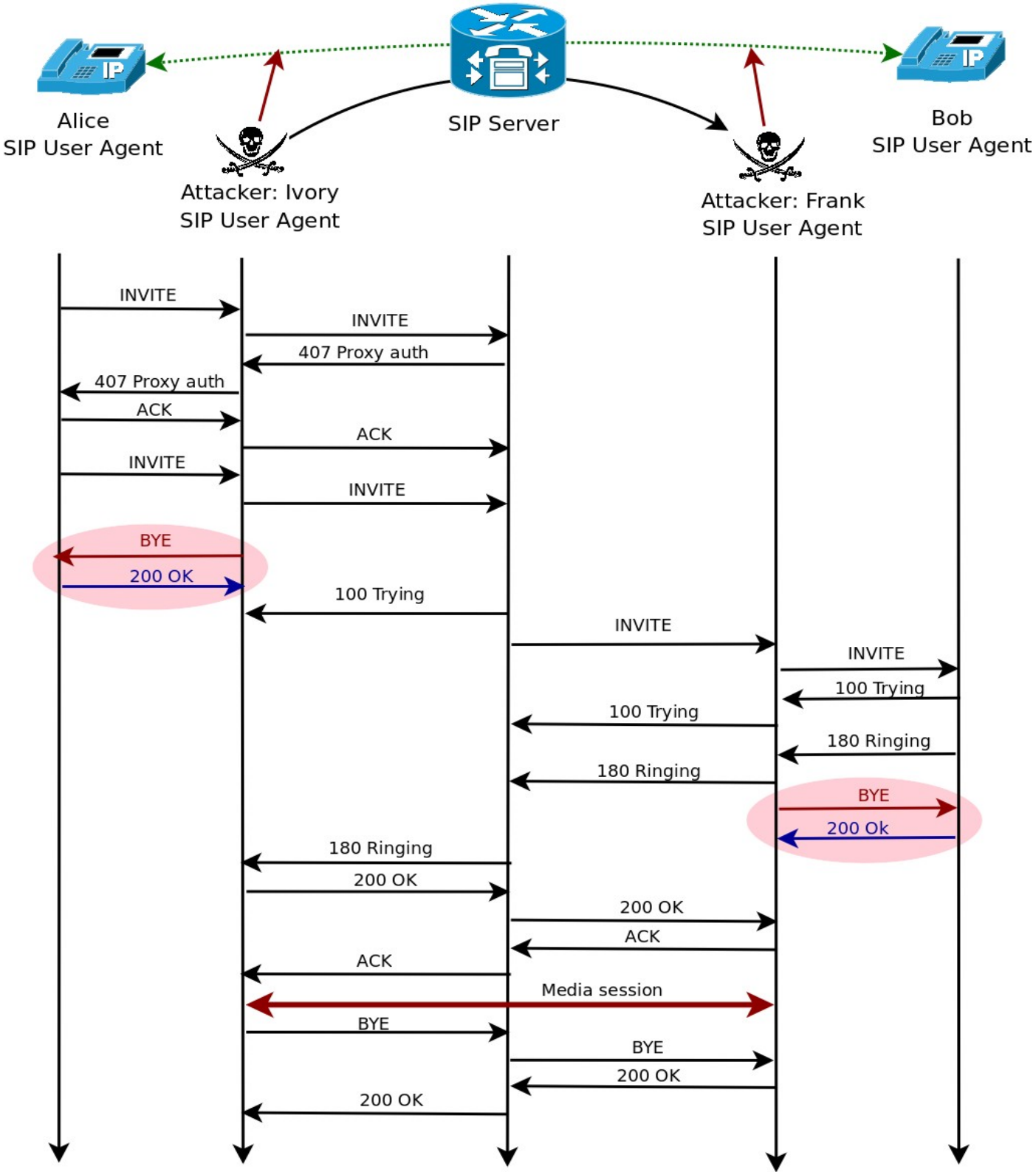
In Asterisk we have found three (minor) deviations from the SIP standard

We have identified three reasons for this:

1) Because the programmers were unaware of the correct standard, or
2) Result of sloppy programming resulting in out of order messages, or
3) Careful deliberation of the programmers to "optimize" the protocol

Call-hijack:

- A client can issue a teardown (BYE) sub-protocol at any time
- Combined with active MitM attack
- Results:
  - Breaks authenticity of the participants – who is really calling?
  - Billing – the attacker sets up an arbitrary call that Alice is billed for
  - VoIP provider can not trust his call logs

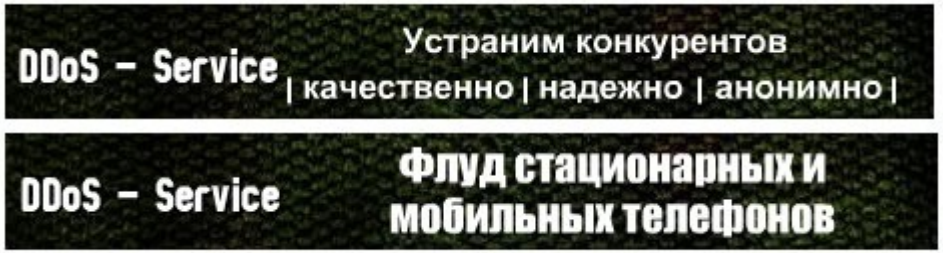

Alice
SIP User Agent
SIP Server
Bob
SIP User Agent
Attacker: Ivory
SIP User Agent
Attacker: Frank
SIP User Agent
INVITE
INVITE
407 Proxy auth
407 Proxy auth
ACK
ACK
INVITE
INVITE
BYE
200 OK
100 Trying
INVITE
INVITE
100 Trying
100 Trying
180 Ringing
180 Ringing
BYE
200 Ok
180 Ringing
200 OK
200 OK
ACK
ACK
Media session
BYE
BYE
200 OK
200 OK

Conclusion

- SIP is a huge and feature rich protocol
  - But, main focus has been on functionality, not security
- More work should have been investigated in the
  - SIP INVITE method and
  - SIP Digest Access Authentication
- Easier to spot security errors when we have a precise specification
  - Our approach can be of aid to protocol designers and implementors
  - Could have prevented the call-hijack attack presented
- Help to raise security awareness and level when using VoIP
  - Lack of security awareness among VoIP providers (upcoming article)
  - Important to use and apply (VoIP) security mechanisms – but are they adequate? (upcoming article)

An afterthought

Russian ad for launching DDoS VoIP attack against an competitor:



The ad scrolls through several messages, including

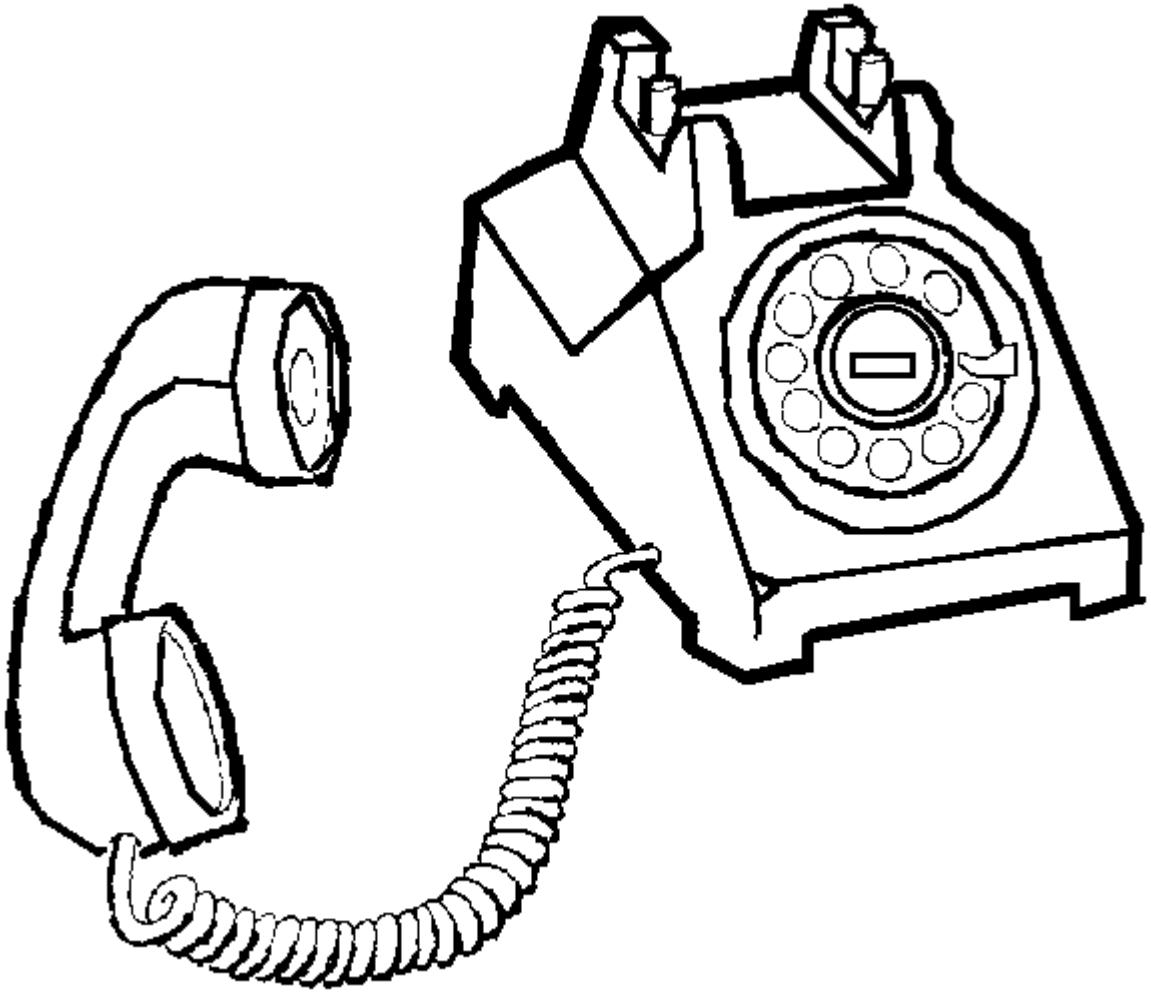
- "Will eliminate competition: high-quality, reliable, anonymous."
- "Flooding of stationary and mobile phones."
- "Pleasant prices: 24-hours start at $80. Regular clients receive significant discounts."
- "Complete paralysis of your competitor/foe."

Flooding of victims phones can be devastating

- SPIT can also turn out to be a major problem

Reference: http://isc.sans.org/diary.html?storyid=5380

Thank you



Project homepage: http://eux2010sec.nr.no