

# SISU

Ein Web-Service zum Testen der  
Sicherheit SIP-basierter Voice-  
over-IP Endgeräte

Jan Seedorf

Stephan Sutardi

DFN Workshop "Sicherheit in vernetzten Systemen"

# Überblick

- Einführung
  - SIP
- Tests
- SISU
- Ergebnisse
- Zusammenfassung

# Einführung

- Signalisierung von VoIP mit SIP
  - ASCII Textnachricht
- Sprachübertragung mit RTP

# SIP – Eigenschaften

- Der momentane Aufenthaltsort des Benutzers kann bestimmt werden.
- Benutzerverfügbarkeit kann überprüft werden
- Bestimmung der zu benutzenden Mediatypen und Parameter
- Aufbau einer Session
- Session Management

# SIP - Nachricht

```
INVITE sip:SIP@134.100.22.62 SIP/2.0
```

Start-Zeile

```
To: sip:SIP@134.100.22.62;  
From: sip:test@171.53.108.116;tag=50962  
Call-ID: 9211@171.53.108.116  
CSeq: 1 INVITE  
Contact: sip:test@134.100.22.62;  
Max-Forwards: 70  
Allow: INVITE, ACK, CANCEL, BYE, NOTIFY  
Content-Type: application/sdp  
Content-Length: 137
```

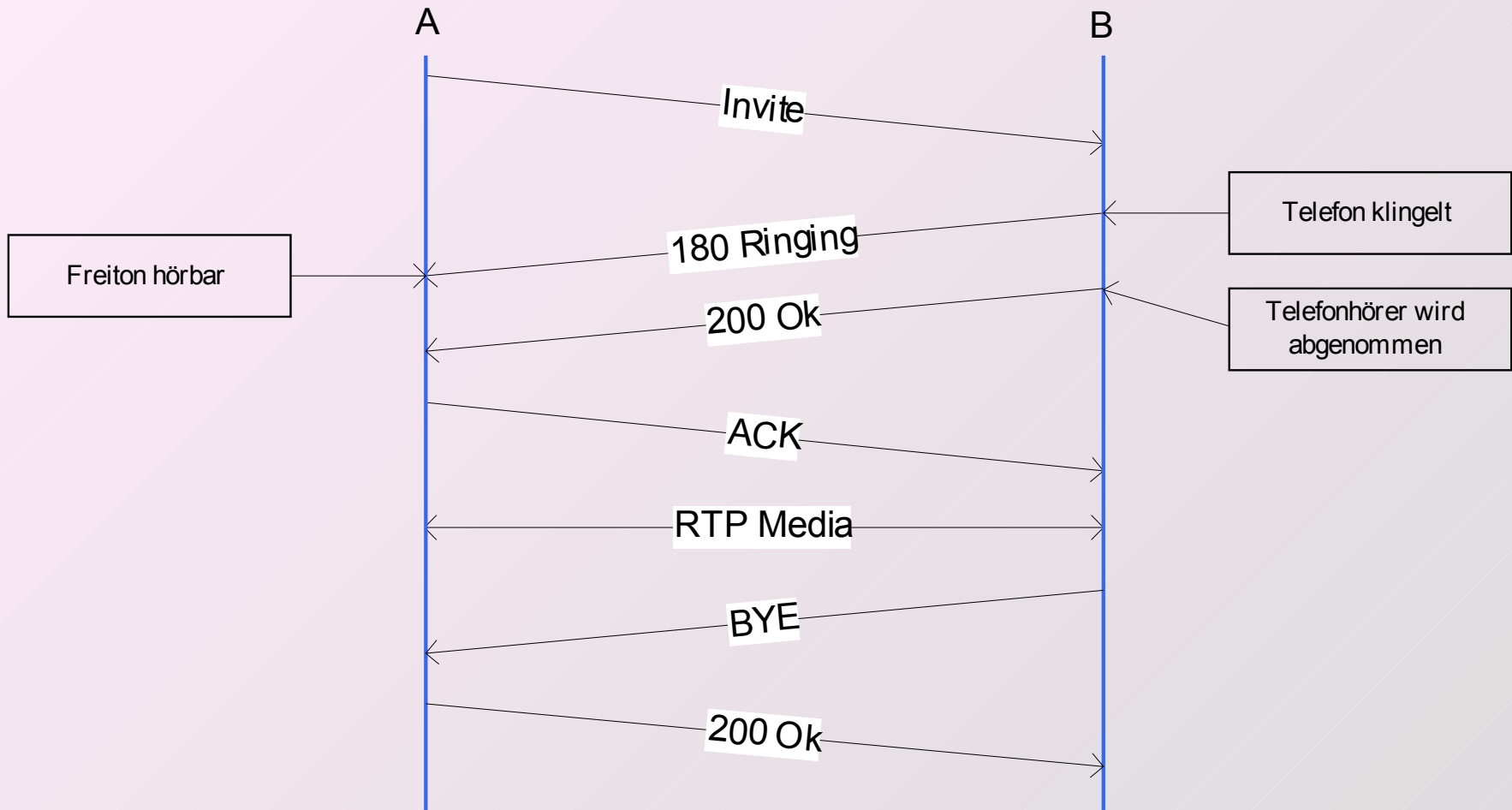
Nachrichtenkopf (SIP)

Leerzeile

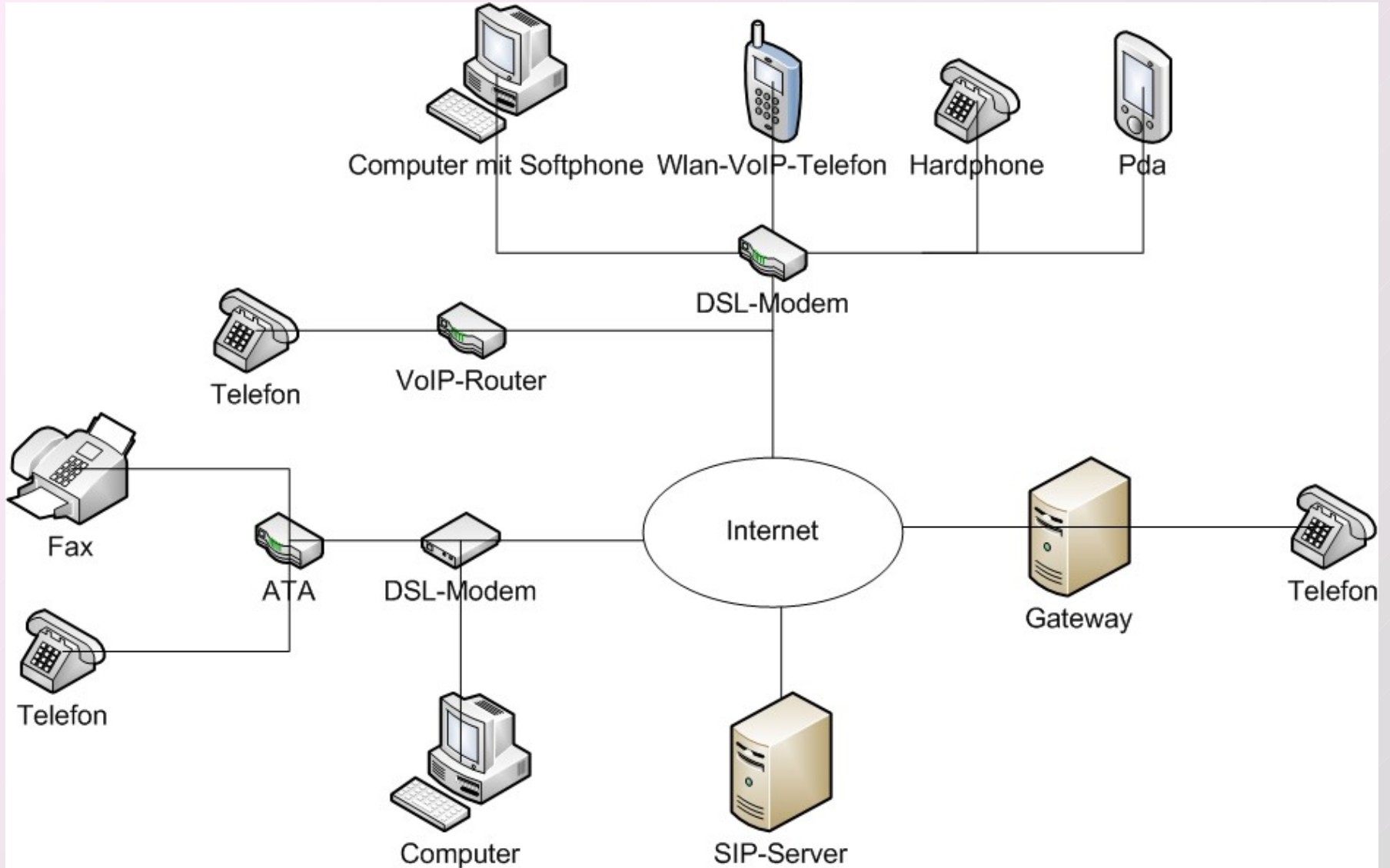
```
v=0  
o=user1 53655765 2353687637 IN IP4  
134.100.22.62  
s=-  
c=IN IP4 134.100.22.62  
t=0 0  
m=audio 5060 RTP/AVP 0  
a=rtpmap:0 PCMU/8000
```

Nachrichtenkörper (SDP)

# Verbindungsaufbau mit SIP



# SIP-Entitäten

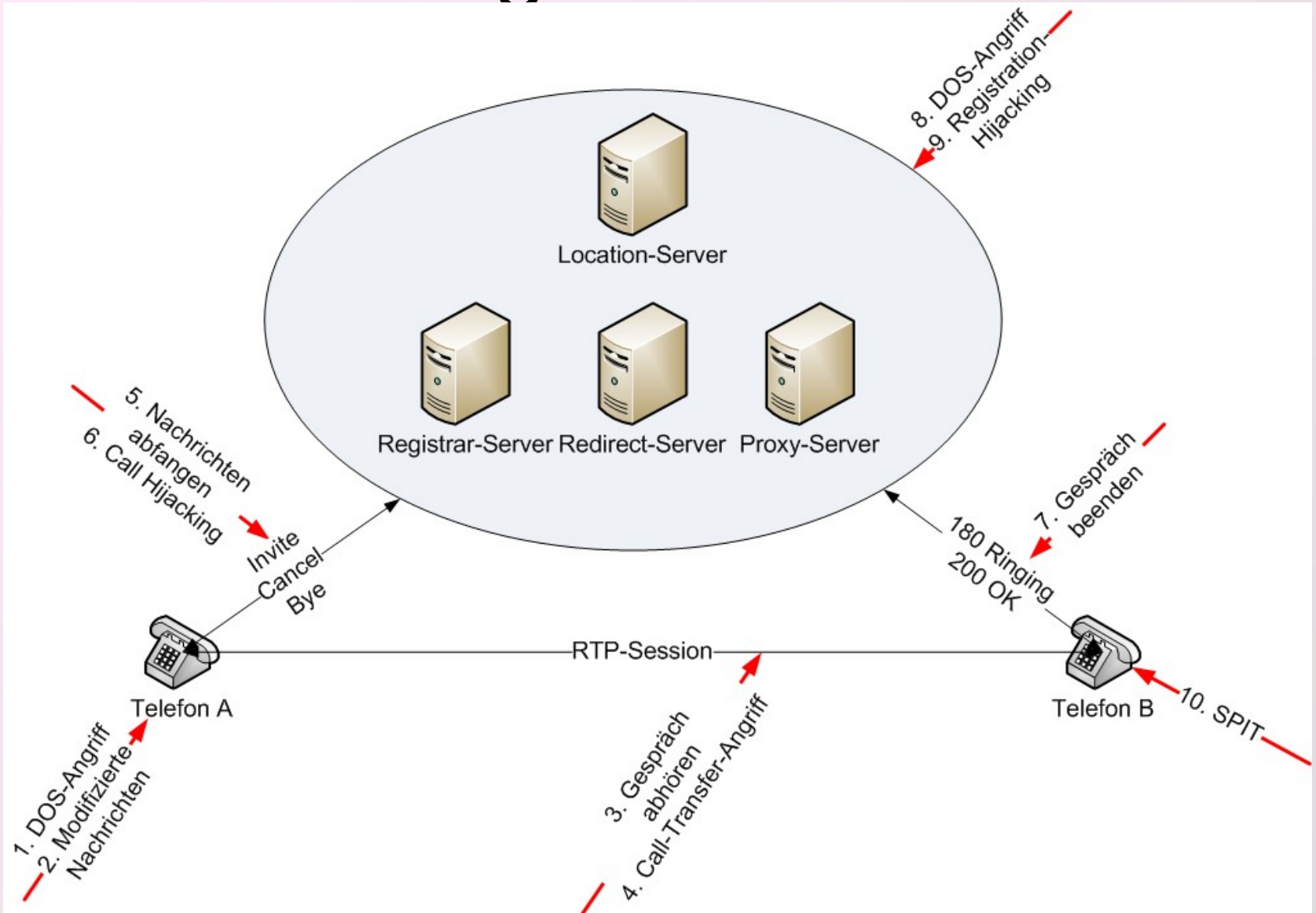


# Tests

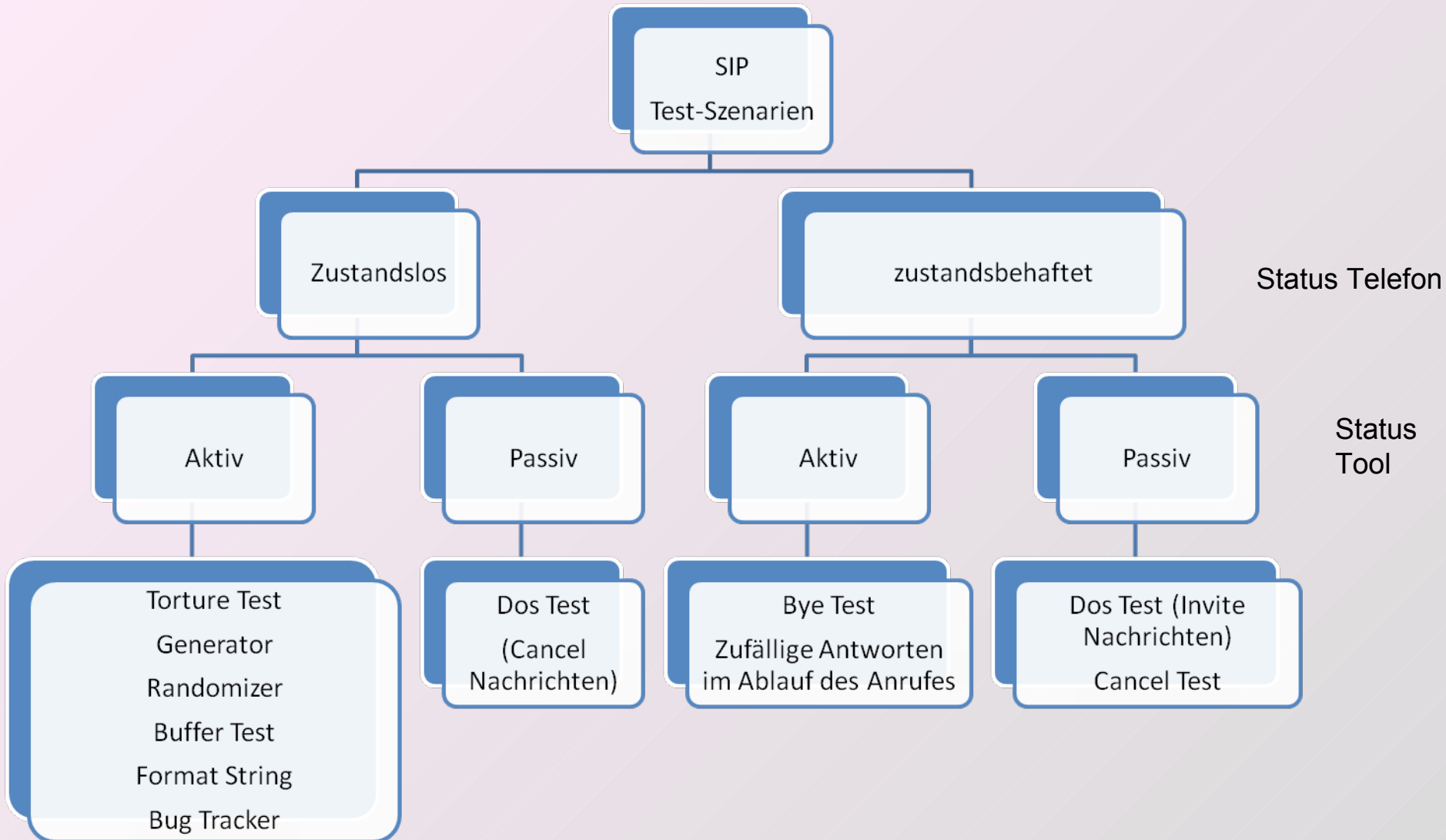
- Arten von Angriffen
- Test-Szenarien



# Angriffs-Arten



# Test-Szenarien (1/3)



# Test-Szenarien (2/3)

- Torture-Test
  - RFC 3261
- Generator
  - Benutzer kann angepasste Invite-Nachrichten verschicken
- Randomizer
  - Erstellt zufällige SIP-Nachrichten
- Buffer-Test
  - Buffer Overflow

# Test-Szenarien (3/3)

- Format String
  - Suche nach einer Format-String-Lücke
- DOS-Test
  - Ziel: Erreichbarkeit einzuschränken
- Cancel-Test
  - Unautorisiert den Verbindungsaufbau zu verhindern
- Bye-Test
  - Unautorisiert die Verbindung zu verhindern

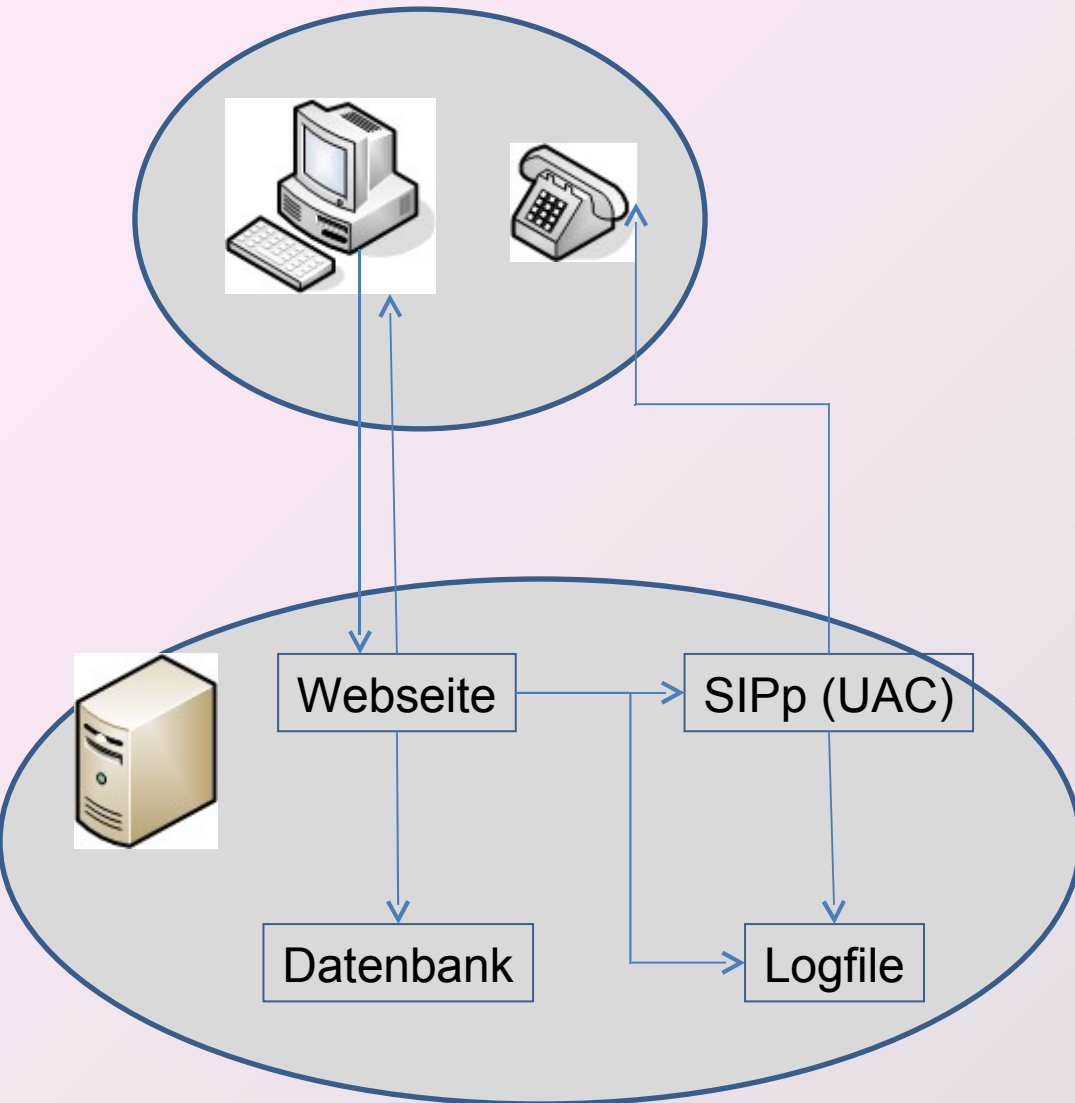
# Entstehung von SISU

- Projekt
- Diplomarbeit
- Ziele:
  - Benutzerfreundlichkeit
  - Hohe Erreichbarkeit
  - Leichtes Integrieren von Tests
  - Breite Palette an Testcases
  - Speichern der Ergebnisse in einer Datenbank

# Einsatzgebiete

- World Wide Web
  - Für jeden Nutzer
- Lokale Netz
  - Für Administratoren

# Architektur



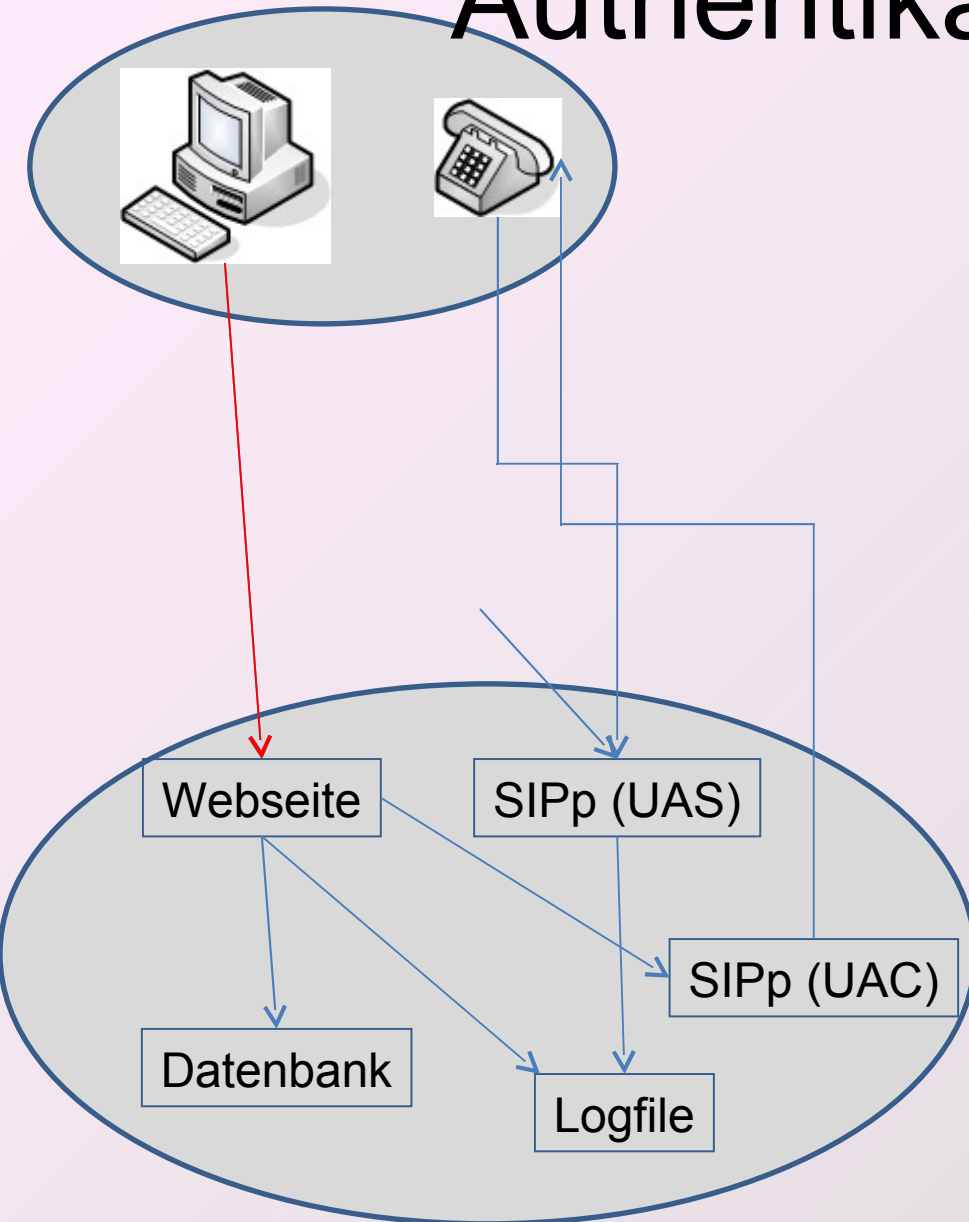
1. Aufruf
2. Ausgabe
3. SIPp aufrufen
4. Testen
5. Speichern
6. Logfile auswerten
7. Datenbank speichern

# Authentikation (1/2)

- Grund:
  - keine ungewollten Tests an Telefonen
- Die Authentifizierung des Telefons erfolgt in neun Schritten



# Authentikation (2/2)



1. SIPp als UAS starten
2. Aufruf der Webseite
3. Phone ruft SIP-URI an
4. Loggen
5. Daten Abrufen
6. SIPp als UAC aufrufen
7. Invite mit zufälliger SIP-URI
8. SIP-URI eingeben
9. Speichern

# SISU - Login



[▶ Home](#) | [Register](#) | [Contact](#)



SIP testing tool

[▶ Login](#)

E-Mail:

Password:

[▶ registration](#)

[▶ The SISU 4 Web Project](#)

[▶ About](#)

# SISU - Main



▶ [Home](#) | [Register](#) | [Contact](#)



Name: 12345678  
IP: 876541

- ▶ Register your Phone
- ▶ Editor IP and Name
- ▶ Former Results

▶ [The SIP Project](#)

▶ [Login](#)

You are log on as stephan  
with the mail a@b.de

▶ [Testmenu](#)

## RFC 4475 SIP Torture Test Messages

- ▶ Parser Test Valid Messages
- ▶ Parser Test Invalid Messages
- ▶ More Tests

## Buffer Overflow

- ▶ Buffer Overflow Test

## Format String

- ▶ Format String Test

## DOS-Test

- ▶ Invite
- ▶ Cancel

## Call Flow

- ▶ Cancel-Test
- ▶ Bye-Test

## Generator

- ▶ Invite-Generator
- ▶ Invite-Generator Netcat -  
Send an Invite and a Cancel Message

## Randomizer

- ▶ Random n-Messages with Netcat
- ▶ Random n-Messages with Netcat  
and own String
- ▶ Random Sequence with Netcat

## Vulnerability Test

- ▶ Found by SISU
- ▶ Found on Bug Tracker
  
- ▶ Normal Invite

# SISU - Testmenu

## ▶ Testmenu

### RFC 4475 SIP Torture Test Messages

- ◆ Parser Test Valid Messages
- ◆ Parser Test Invalid Messages
- ◆ More Tests

### Buffer Overflow

- ◆ Buffer Overflow Test

### Format String

- ◆ Format String Test

### DOS-Test

- ◆ Invite
- ◆ Cancel

### Call Flow

- ◆ Cancel-Test
- ◆ Bye-Test

### Generator

- ◆ Invite-Generator
- ◆ Invite-Generator Netcat -  
Send an Invite and a Cancel Message

### Randomizer

- ◆ Random n-Messages with Netcat
- ◆ Random n-Messages with Netcat  
and own String
- ◆ Random Sequence with Netcat

### Vulnerability Test

- ◆ Found by SISU
- ◆ Found on Bug Tracker
  
- ◆ Normal Invite

# SISU – Buffer-Test-Menu

## Buffer Test

### 500 same character ->"a"s

- ◆ after the Invite (Header) using Netcat
- ◆ Branch (Header) using Netcat
- ◆ Tag (Header) using Netcat
- ◆ the Call-Id (Header) using Netcat
- ◆ as a Cseq Value using Netcat

### 2000 same character ->"a"s

- ◆ after the Invite (Header) using Netcat
- ◆ Branch (Header) using Netcat
- ◆ Tag (Header) using Netcat
- ◆ the Call-Id (Header) using Netcat
- ◆ as a Cseq Value using Netcat

### 16144 same character ->"a"s

- ◆ after the Invite (Header) using Netcat
- ◆ after Branch (Header) using Netcat
- ◆ after Tag (Header) using Netcat
- ◆ after the Call-Id (Header) using Netcat
- ◆ as a Cseq Value using Netcat

# SISU - Generator

## SIP - Invite

```
INVITE sip:[service]@[remote_ip]:[remote_port]
SIP/2.0
Via: SIP/2.0/[transport]
[local_ip]:[local_port];branch=[branch]
From: crash
<sip:crash@[local_ip]:[local_port]>;tag=[call_number]
To: crash
<sip:[service]@[remote_ip]:[remote_port]>[peer_tag]
Call-ID: [call_id]
CSeq: 1 INVITE
```

## SDP

```
v=0
o=user1 53655765 2353687637 IN IP[local_ip_type]
[local_ip]
s=-
c=IN IP[media_ip_type] [media_ip]
t=0 0
m=audio [media_port] RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

## ACK

```
From: crash
<sip:crash@[local_ip]:[local_port]>;tag=[call_number]
To: crash
<sip:[service]@[remote_ip]:[remote_port]>[peer_tag]
Call-ID: [call_id]
CSeq: 1 ACK
Contact: sip:crash@[local_ip]:[local_port]
Max-Forwards: 70
Content-Length: 0
```

## BYE

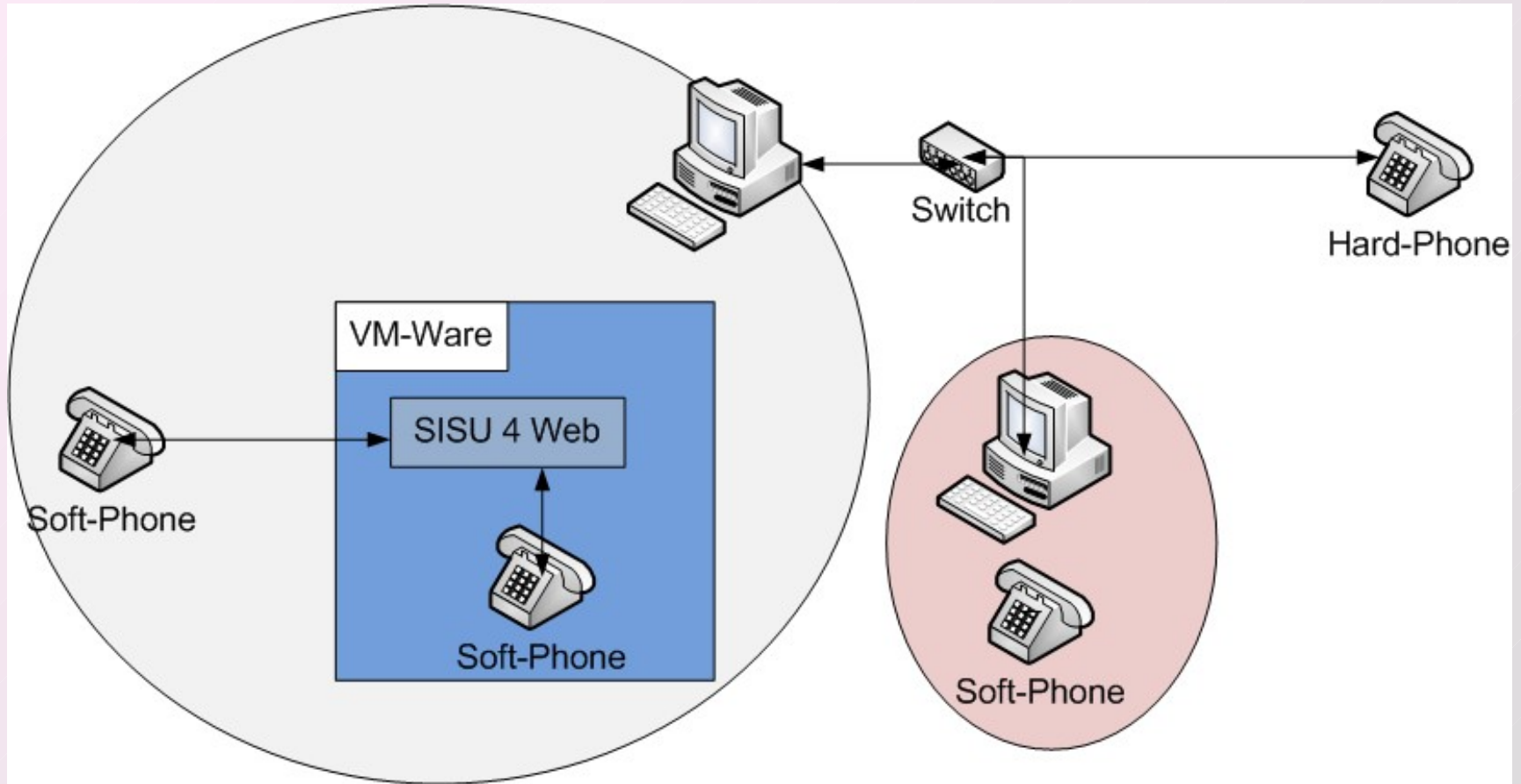
```
BYE sip:[service]@[remote_ip]:[remote_port]
SIP/2.0
Via: SIP/2.0/[transport]
[local_ip]:[local_port];branch=[branch]
From: crash
<sip:crash@[local_ip]:[local_port]>;tag=[call_number]
To: crash
<sip:[service]@[remote_ip]:[remote_port]>[peer_tag]
Call-ID: [call_id]
```

Send 2 Phone

# Testen und Ergebnisse

- Testumgebung
- Ergebnisse
- Diskussion

# Testumgebung





# Zusammenfassung der Testergebnisse

- Syntax
  - die Anzahl der bestandenen Tests (das Telefon antwortete mit der erwarteten SIP-Nachricht),
  - die Anzahl an nicht bestandenen Tests (das Telefon antwortete nicht mit der erwarteten SIP-Nachricht)
  - und die Anzahl an Abstürzen

# Ergebnisse - Hardphones

Testart	H1	H2	H3	H4
Torture Test (48)	(12/35/1)	(16/32/0)	(19/29/0)	(15/33/0)
Transaktion Bye (6)	(2/4/0)	(4/2/0)	(2/4/0)	(2/4/0)
Transaktion Cancel (8)	(6/2/0)	(4/4/0)	(8/0/0)	(8/0/0)
Stress-Test (2)	(1/1/0)	(1/1/0)	(1/1/0)	(1/1/0)
Buffer-Overflow (15)	(7/0/8)	(15/0/0)	(15/0/0)	(15/0/0)
Format-String (30)	(30/0/0)	(30/0/0)	(30/0/0)	(30/0/0)



# Diskussion

- Torture-Test
  - Schwächen der Implementation des SIP-Standards
- Transaktionstest (Bye und Cancel)
  - Bye: Nur ein Hardphone war ziemlich erfolgreich
  - Cancel: bessere Überprüfung als bei Bye
- DOS:
  - Invite: alle hatten Problem
  - Cancel: erfolglos
- Buffer und Format-String-Test
  - In der Regel waren alle Telefone resistent gegen diese Angriffe

# Fehlermeldungen



## **Fehler: Netzwerk-Zeitüberschreitung**

---

Der Server unter 10.10.10.14 braucht zu lange, um eine Antwort zu senden.

---

- Die Website könnte vorübergehend nicht erreichbar sein, versuchen Sie es bitte später nochmals.
- Wenn Sie auch keine andere Website aufrufen können, überprüfen Sie bitte die Netzwerk-/Internetverbindung.
- Wenn Ihr Computer oder Netzwerk von einer Firewall oder einem Proxy geschützt wird, stellen Sie bitte sicher, dass Firefox auf das Internet zugreifen darf.

Nochmals versuchen

# Fehlermeldungen

**hat ein Problem festgestellt und muss beendet**

Falls Sie Ihre Arbeit noch nicht gespeichert hatten, können Daten möglicherweise verloren gegangen sein.

**Dieses Problem bitte auch an Microsoft berichten.**

Ein Problembenicht, den Sie uns senden können, wurde erstellt. Wir werden diesen Bericht vertraulich und anonym bearbeiten.

Um zu sehen, welche Daten Ihr Bericht enthält, [klicken Sie hier](#).

Problembenicht senden

Nicht senden

# Zusammenfassung

- Einführung von SIP
- Test-Szenarien
- Vorstellung von SISU
- Test von mehreren SIP-Implementationen
- Darstellung der Ergebnisse

Fragen ?



# Kontakt

Jan Seedorf  
NEC Laboratories Europe  
Kurfürstenanlage 36  
69115 Heidelberg  
[jan.seedorf@nw.neclab.eu](mailto:jan.seedorf@nw.neclab.eu)

Stephan Sutardi  
Sicherheit in Verteilten Systemen  
Univ. Hamburg, Dept. Informatik  
Vogt-Kölln-Str. 30  
22527 Hamburg  
[sutardi@gmail.com](mailto:sutardi@gmail.com)