

Asterisk Grundkonfiguration



Über die Konfigurationsdateien im Verzeichnis **/etc/asterisk/** wird das Verhalten unserer Asteriskinstallation beeinflusst.

Wir bearbeiten nun die einzelnen Konfigurationsdateien, die wir der Reihe nach durchgehen. Wie gewohnt bearbeiten wir die mitgelieferten Vorgabedateien und versehen diese mit entsprechenden Kommentaren. In aller Regel werden *Kommentare* in den Dateien mit einem Semikolon **;** markiert. Somit können wir dann bei Bedarf die reinen Konfigurationsdaten mittels eines **egrep**-Aufrufes gerafft ausgeben.

logger.conf

In der Datei **logger.conf** werden folgende Daten der Log-Einträge unseres Asterisk-Servers definiert:

- Art
- Format
- Umfang

Bei der Erstinstallation bzw. bei umfangreichen Test kann es hilfreich sein, die „*Gesprächigkeit*“ unseres Asterisk-Servers zu erhöhen. Hierzu aktivieren wir die bereits vorhandene Option:

```
full => notice,warning,error,debug,verbose
```

Für den Regelbetrieb wird sich unsere Konfiguration aber auf nachfolgende Zeilen beschränken:

```
# egrep -v '(^s+;|^;|^$)' /etc/asterisk/logger.conf
```

```
[general]
[logfiles]
console => notice,warning,error
messages => notice,warning,error
```

modules.conf

Den Modullademechanismus von Asterisk bestimmen wir über die Konfigurationsdatei **modules.conf**.

```
# egrep -v '(^s+;|^;|^$)' /etc/asterisk/modules.conf
```

```
[modules]
autoload=yes
preload => func_strings.so
noload => pbx_gtkconsole.so
load => res_musiconhold.so
noload => chan_alsa.so
noload => chan_console.so
```

sip.conf

general

Der Kanaltyp **SIP**¹⁾ wird über die zentrale Konfigurationsdatei **sip.conf** definiert. Im ersten Schritt definieren wir erst einmal den Voicemailserver-Port und auch aktivieren die deutschen Sprachprompts.

```
# vim /etc/asterisk/sip.conf
```

```
; Django 26.11.2010 zentrale Rufnummer für den Voicemailserver
vmexten=60
```

```
; Django 26.11.2010 deutsche Promts für alle ausgeben
language=de
```

Die Gesamt-Konfiguration beschränkt sich demnach erst einmal auf:

```
# egrep -v '^(^\s+;|^;|^$)' /etc/asterisk/sip.conf
```

```
[general]
context=default                ; Default context for incoming calls
allowoverlap=no                ; Disable overlap dialing support. (Default
is yes)
udpbindaddr=0.0.0.0            ; IP address to bind UDP listen socket to
(0.0.0.0 binds to all)
tcpenable=no                   ; Enable server for incoming TCP connections
(default is no)
tcpbindaddr=0.0.0.0           ; IP address for TCP server to bind to
(0.0.0.0 binds to all interfaces)
srvlookup=yes                  ; Enable DNS SRV lookups on outbound calls
vmexten=60
language=de
[authentication]
[basic-options](!)             ; a template
    dtmfmode=rfc2833
    context=from-office
    type=friend
[natted-phone](!,basic-options) ; another template inheriting basic-
options
```

```
        nat=yes
        directmedia=no
        host=dynamic
[public-phone](!,basic-options) ; another template inheriting basic-
options
        nat=no
        directmedia=yes
[my-codecs](!) ; a template for my preferred codecs
        disallow=all
        allow=ilbc
        allow=g729
        allow=gsm
        allow=g723
        allow=ulaw
[ulaw-phone](!) ; and another one for ulaw-only
        disallow=all
        allow=ulaw
```

phones

Im zweiten Anlauf definieren wir uns nun unsere SIP-Endgeräte. Hierzu legen uns nun pro Endstelle einen Eintrag, nach folgendem Muster an.

```
[ID]
callerid="Name der Endstelle"
type="User-Type"
username="Username"
secret="Passwort"
mailbox="Mailboxrufnummer"
host="Host-Type"
domain="Registrar"
canreinvite="RTP-Medienstrom Definition"
callgroup="Call-Gruppe"
pickupgroup="Pickup-Gruppe"
```

Im einzelnen haben die Parameter folgende Bedeutung:

[ID]

ID des SIP-Teilnehmers mit welcher dieser Teilnehmer beim Asterisk-Server bekannt ist. Neben einer reinen numerischen Angabe kann man auch mit Namen und/oder alphanumerischen Kombinationen arbeiten. Der Übersichtlichkeit halber empfiehlt es sich, die ID analog der Rufnummer zu verwenden.

callerid

Überschreibt vom Gerät gesendete die Caller-ID. Diese CallerID besteht in der Regel aus zwei Teilen, dem CallerIDName und CallerIDNum. Mit Hilfe der spitzen Klammern <>, welche die Rufnummer umklammert wird die Nummer vom Namen abgetrennt.

type

Über den **type** wird festgelegt, in welcher Richtung der Teilnehmer Verbindungen aufbauen kann/darf. Folgende Richtungen sind festlegbar:

```
<uml w=210> ( \n\nAsterisk - Server\n\n ) -left-|> :Peer: :User: -|> ( \n\nAsterisk - Server\n\n ) :Friend: -up-|> ( \n\nAsterisk - Server\n\n ) :Friend: <|-up- ( \n\nAsterisk - Server\n\n ) </uml>
```

Ein **user** kann also **_nur_** anrufen, ein **peer** **_nur_** angerufen werden und ein **friend** darf beides.

username

Username, mit dem sich das Endgerät bei der Anmeldung beim Asterisk-Server meldet.

secret

Passwort, welches das Endgerät zusammen mit dem vorgenannten Username bei der Anmeldung am Asterisk-Server verwendet.

mailbox

Definition der Mailbox-Nummer die dem Endgerät zugeordnet werden soll. Dieser Eintrag wird auch für die MWI²⁾, also der Anzeigen von neuen Nachrichten am Endgerät via LED oder Displaynachricht.

host

Festlegung des Hostnamen bzw. Adresse des Endgerätes/Partners. **dynamic** gibt man an, wenn der Hostname bzw. die IP-Adresse des Hosts nicht bekannt ist, da diese Daten dynamisch vergeben werden.

domain

Festlegung des Domainnamens bzw. der IP-Adresse unseres Asterisk-Servers.

canreinvite

Normalerweise versucht der Asterisk-Server für den Mediendatenstrom (RTP³⁾) den direkten Weg von Teilnehmer **A** zum Teilnehmer **B** einzuschlagen. Soll der Asterisk-Server auf Eingaben während des Gespräches reagieren, oder wenn sich die Geräte in einem „**geNATteten**“ Netzwerk befinden, so muss

der Server als Proxy fungieren. Mit der Option **canreinvite** kann nun der RTP-Datenstrom beeinflusst werden.

- **yes** (default) Normale Einstellung. Asterisk versucht, den direkten RTP-Datenstrom zwischen den beiden Partnern zu etablieren.
- **nonat** Den RTP-Medienstrom nur umleiten, wenn sich die Teilnehmer nicht hinter einem NAT befinden und dies für den Asterisk-Server erkennbar ist.
- **update** Zum Umleiten sollen anstatt INVITE- UPDATE-Pakete verwendet werden.

callgroup

Definition einer oder mehrerer Call-Groups für das Endgerät. Mehrere Gruppen werden dabei mittels Komma und eine Reihe von Gruppen mit einem Minus - getrennt. Es können insgesamt 64 Gruppen, von **0** bis **63** definiert werden.

```
callgroup=6,13-15 ; setzt die Gruppen 6, 13, 14 und 15
```

pickupgroup

Definiert die Pickup-Gruppe(n) für das Endgerät, über die in Verbindung mit dem vorgenannten **callgroup**-Definitionen Anrufe herangeholt (*pick-up*) erlaubt sein soll.

```
pickupgroup=6,13-15 ; Pickup für die Gruppen 6, 13, 14 und 15  
definieren
```

Somit ergibt sich für unsere grundlegende Erstinstallation folgende Konfiguration.

```
# egrep -v '^(^s+;|^;|^$)' /etc/asterisk/sip.conf
```

```
[general]  
context=default ; Default context for incoming calls  
allowoverlap=no ; Disable overlap dialing support. (Default  
is yes)  
udpbindaddr=0.0.0.0 ; IP address to bind UDP listen socket to  
(0.0.0.0 binds to all)  
tcpenable=no ; Enable server for incoming TCP connections  
(default is no)  
tcpbindaddr=0.0.0.0 ; IP address for TCP server to bind to  
(0.0.0.0 binds to all interfaces)  
srvlookup=yes ; Enable DNS SRV lookups on outbound calls  
vmexten=60  
language=de  
[authentication]  
[basic-options](!) ; a template  
 dtmfmode=rfc2833  
 context=from-office
```

```
        type=friend
[natted-phone](!,basic-options) ; another template inheriting basic-
options
    nat=yes
    directmedia=no
    host=dynamic
[public-phone](!,basic-options) ; another template inheriting basic-
options
    nat=no
    directmedia=yes
[my-codecs](!) ; a template for my preferred codecs
    disallow=all
    allow=ilbc
    allow=g729
    allow=gsm
    allow=g723
    allow=ulaw
[ulaw-phone](!) ; and another one for ulaw-only
    disallow=all
    allow=ulaw
[Testaccount_11]
callerid=Wohnzimmer <11>
type=friend
username=11
secret=12qwasyx
mailbox=11
host=dynamic
domain=192.168.10.40
canreinvite=no
language=de
callgroup=1
pickupgroup=1
[12]
callerid=Esszimmer <12>
type=friend
username=12
secret=yaq12wsx
mailbox=12
host=dynamic
domain=192.168.10.40
canreinvite=no
callgroup=1
pickupgroup=1
[13]
callerid=Diele <13>
type=friend
username=13
secret=xsw21qay
mailbox=13
host=dynamic
domain=192.168.10.40
```

```
canreinvite=no
callgroup=1
pickupgroup=1
[14]
callerid=Keller <14>
type=friend
username=14
secret=2lwqsaxy
mailbox=14
host=dynamic
domain=192.168.10.40
canreinvite=no
callgroup=1
pickupgroup=1
```

Sollte der Server noch nicht gestartet worden sein, holen wir dies nun nach.

```
# service asterisk start
```

Anschließend verbinden wir uns mit der Asterisk-Konsole mit folgendem Aufruf:

```
# asterisk -r
```

```
Verbosity is at least 3
asterisk*CLI>
```

Unsere zuvor definierten SIP-Teilnehmer können wir uns nun mit der Abfrage **sip show peers** anzeigen lassen.

```
asterisk*CLI> sip show peers
Name/username      Host              Dyn
Forcerport ACL Port   Status
12/12              (Unspecified)    D
0      Unmonitored
13/13              (Unspecified)    D
0      Unmonitored
14/14              (Unspecified)    D
0      Unmonitored
Testaccount_11/11 (Unspecified)    D
0      Unmonitored
4 sip peers [Monitored: 0 online, 0 offline Unmonitored: 0 online, 4
offline]
```

Sobald sich das erste SIP-Endgerät am Asterisk-Server erfolgreich angemeldet hat, wird uns dies auf der Konsole entsprechend gemeldet.

```
-- Registered SIP '14' at 192.168.10.61 port 5060
```

Eine erneute Abfrage der peers zeigt nun den angemeldeten Teilnehmer.

```
asterisk*CLI> sip show peers
Name/username           Host                               Dyn
Forcerport ACL Port     Status
12/12                   (Unspecified)                    D
0      Unmonitored
13/13                   (Unspecified)                    D
0      Unmonitored
14/14                   192.168.10.51                    D
5061    Unmonitored
Testaccount_11/11      (Unspecified)                    D
0      Unmonitored
4 sip peers [Monitored: 0 online, 0 offline Unmonitored: 1 online, 3
offline]
```

Die Details unseres Users 14 können wir uns mit Hilfe des Befehls **sip show peer** ausgeben lassen.

```
asterisk*CLI> sip show peer 14

* Name      : 14
Secret      : <Set>
MD5Secret   : <Not set>
Remote Secret: <Not set>
Context     : default
Subscr.Cont.: <Not set>
Language    : de
AMA flags   : Unknown
Transfer mode: open
CallingPres : Presentation Allowed, Not Screened
Callgroup   : 1
Pickupgroup : 1
MOH Suggest :
Mailbox     : 14
VM Extension : 60
LastMsgsSent : 32767/65535
Call limit  : 0
Max forwards : 0
Dynamic     : Yes
Callerid    : "Keller" <14>
MaxCallBR   : 384 kbps
Expire      : 3533
Insecure    : no
Force rport : No
ACL         : No
DirectMedACL : No
T.38 support : No
T.38 EC mode : Unknown
T.38 MaxDtgrm: -1
DirectMedia : No
PromiscRedir : No
User=Phone  : No
```

```
Video Support: No
Text Support : No
Ign SDP ver  : No
Trust RPID   : No
Send RPID    : No
Subscriptions: Yes
Overlap dial : No
DTMFmode     : rfc2833
Timer T1     : 500
Timer B      : 32000
ToHost       :
Addr->IP     : 192.168.10.51:5061
Defaddr->IP  : (null)
Prim.Transp. : UDP
Allowed.Trsp : UDP
Def. Username: 14
SIP Options  : (none)
Codecs       : 0x80000008000e (gsm|ulaw|alaw|h263|testlaw)
Codec Order  : (none)
Auto-Framing : No
100 on REG   : No
Status       : Unmonitored
Useragent    : Zoiper rev.5415
Reg. Contact :
sip:14@192.168.10.51:5061;rinstance=2d6a419d01d09ccc;transport=UDP
Qualify Freq : 60000 ms
Sess-Timers  : Accept
Sess-Refresh : uas
Sess-Expires : 1800 secs
Min-Sess     : 90 secs
RTP Engine   : asterisk
Parkinglot   :
Use Reason   : No
```

iax.conf

general

Der Kanaltyp **IAX**⁴⁾ wird über die zentrale Konfigurationsdatei **iax.conf** definiert. **IAX** ist ein Protokoll, welches speziell für Asterisk entwickelt und genutzt wird. Es können damit sowohl Verbindungen zwischen einzelnen Asterisk-Servern als auch die Kommunikation zu Endgeräten ermöglicht werden.

Als aktuelle Version findet die Version 2 (**IAX2**) Verwendung.

Wie auch schon bei der Konfiguration bei der **sip.conf**, aktivieren wir im ersten Schritt auch hier die deutschen Sprachprompts.

```
# vim /etc/asterisk/iax.conf
```

```
; Django 01.12.2010 deutsche Prompts für alle ausgeben
```

```
language=de
```

Anschließend definieren wir noch, dass die Überprüfung der Anrufberechtigung im eigenen Netzwerk, deaktiviert werden soll. Somit ist sichergestellt, dass unser [Zoiper-Client](#) mit unserem Asterisk-Server kommunizieren kann.

```
# vim /etc/asterisk/iax.conf
```

```
; Django 01.12.2010  
calltokenoptional=192.168.10.0/255.255.255.0
```

Die Gesamt-Konfiguration beschränkt sich demnach erst einmal auf:

```
# egrep -v '^(^s+|^;|^$)' /etc/asterisk/iax.conf
```

```
[general]  
language=de  
bandwidth=low  
disallow=lpc10 ; Icky sound quality... Mr. Roboto.  
jitterbuffer=no  
forcejitterbuffer=no  
autokill=yes  
calltokenoptional=192.168.10.0/255.255.255.0  
[guest]  
type=user  
context=default  
callerid="Guest IAX User"  
[iaxtel]  
type=user  
context=default  
auth=rsa  
inkeys=iaxtel  
[iaxfwd]  
type=user  
context=default  
auth=rsa  
inkeys=freeworlddialup  
[demo]  
type=peer  
username=asterisk  
secret=supersecret  
host=216.207.245.47
```

phones

Im zweiten Anlauf definieren wir uns nun unsere SIP-Endgeräte. Hierzu legen uns nun pro Endstelle einen Eintrag, nach folgendem Muster an.

```
[ID]
```

```
callerid="Name der Endstelle"  
type="User-Type"  
username="Username"  
secret="Passwort"  
mailbox="Mailboxrufnummer"  
host="Host-Type"  
transfer="IAX-Medienstrom Definition"  
language="Sprachewahl der Meldungen"
```

Im einzelnen haben die Parameter folgende Bedeutung:

[ID]

ID des IAX-Teilnehmers mit welcher dieser Teilnehmer beim Asterisk-Server bekannt ist. Neben einer reinen numerischen Angabe kann man auch mit Namen und/oder alphanumerischen Kombinationen arbeiten. Der Übersichtlichkeit halber empfiehlt es sich, die ID analog der Rufnummer zu verwenden.

callerid

Überschreibt vom Gerät gesendete die Caller-ID. Diese CallerID besteht in der Regel aus zwei Teilen, dem CallerIDName und CallerIDNum. Mit Hilfe der spitzen Klammern <>, welche die Rufnummer umklammert wird die Nummer vom Namen abgetrennt.

type

Über den **type** wird festgelegt, in welcher Richtung der Teilnehmer Verbindungen aufbauen kann/darf. Folgende Richtungen sind festlegbar:

- Asterisk ↔ friend
- Asterisk ⇒ peer
- Asterisk ⇐ user

Ein **user** kann also *_nur_* anrufen, ein **peer** *_nur_* angerufen werden und ein **friend** darf beides.

username

Username, mit dem sich das Endgerät bei der Anmeldung beim Asterisk-Server meldet.

secret

Passwort, welches das Endgerät zusammen mit dem vorgenannten Username bei der Anmeldung am Asterisk-Server verwendet.

mailbox

Definition der Mailbox-Nummer die dem Endgerät zugeordnet werden soll. Dieser Eintrag wird auch für die MWI⁵⁾, also der Anzeigen von neuen Nachrichten am Endgerät via LED oder Displaynachricht.

host

Festlegung des Hostnamen bzw. Adresse des Endgerätes/Partners. **dynamic** gibt man an, wenn der Hostname bzw. die IP-Adresse des Hosts nicht bekannt ist, da diese Daten dynamisch vergeben werden.

transfer

Ähnlich wie schon bei der Konfiguration der SIP-Channels, kann hier definiert werden, ob der Mediendatenstrom direkt zwischen Clients erfolgen darf/soll oder eben nicht.

language

Über diese Option kann die Sprache der Meldungen beim Client unabhängig von der globalen Definition geändert werden.

Somit ergibt sich für unsere IAX-Erstinstallation folgende Konfiguration.

```
# egrep -v '^(^s+;|^;|^$)' /etc/asterisk/iax.conf
```

```
[general]
language=de
bandwidth=low
disallow=lpcl0                ; Icky sound quality... Mr. Roboto.
jitterbuffer=no
forcejitterbuffer=no
autokill=yes
calltokenoptional=192.168.10.0/255.255.255.0
[guest]
type=user
context=default
callerid="Guest IAX User"
[iaxtel]
type=user
context=default
auth=rsa
inkeys=iaxtel
[iaxfwd]
type=user
```

```
context=default
auth=rsa
inkeys=freeworlddialup
[demo]
type=peer
username=asterisk
secret=supersecret
host=216.207.245.47
[33]
callerid="Zoiper 33 - IAX Phone <33>"
type=friend
username=33
secret=Z|pf3gl@t$cha!
mailbox=60
host=dynamic
transfer=no
language=de
```

Sollte der Server noch nicht gestartet worden sein, holen wir dies nun nach.

```
# service asterisk start
```

Lief der Telephonieserver bereit, starten wir diesen einfach durch.

```
# service asterisk restart
```

Anschließend verbinden wir uns mit der Asterisk-Konsole mit folgendem Aufruf:

```
# asterisk -r
```

```
Verbosity is at least 3
asterisk*CLI>
```

Unseren gerade eben definierten IAX-Teilnehmer können wir uns nun mit der Abfrage **iax2 show peers** anzeigen lassen.

```
asterisk*CLI> iax2 show peers
Name/Username      Host                Mask                Port                Status
33/33              192.168.10.51      (D) 255.255.255.255  4570
Unmonitored
demo/asterisk      216.207.245.47     (S) 255.255.255.255  4569
Unmonitored
2 iax2 peers [0 online, 0 offline, 2 unmonitored]
asterisk*CLI>
```

Sobald sich der IAX-Client am Asterisk-Server erfolgreich registriert hat, wird uns dies auf der Konsole entsprechend gemeldet.

```
-- Registered IAX2 '33' (AUTHENTICATED) at 192.168.10.51:4570
```

Die Details unseres Users 33 können wir uns mit Hilfe des Befehls **iax2 show peer** ausgeben lassen.

```
* Name      : 33
Secret     : <Set>
Context   :
Parking lot :
Mailbox    : 60
Dynamic    : Yes
Callnum limit: 0
Calltoken req: No
Trunk      : No
Encryption : No
Callerid   : "Zoiper 33 - IAX Phone" <33>
Expire     : 31
ACL        : No
Addr->IP   : 232.44.190.178 Port 0
Defaddr->IP : 0.0.0.0 Port 0
Username   : 33
Codecs     : 0x703 (g723|gsm|g729|speex|ilbc)
Codec Order : (none)
Status     : Unmonitored
Qualify    : every 60000ms when OK, every 10000ms when UNREACHABLE
(sample smoothing Off)
```

1)

Session **I**nitiation **P**rotokoll

2) , 5)

Message **W**aiting **I**ndication

3)

Real **T**ime **P**rotocol

4)

Inter **A**sterisk **eX**change

From:
<https://dokuwiki.nausch.org/> - **Linux - Wissensdatenbank**

Permanent link:
<https://dokuwiki.nausch.org/doku.php/voip:asterisk18:config>

Last update: **20.04.2018 08:59.**

