

[In English](#) ----- [På svenska](#) ----- [En español](#) ----- [En français](#)

Wichtiger Hinweis: Leider verfälschen die Google-Übersetzungen die Quellcodes. Bitte öffnet den Quellcode in einem anderen Fenster, wie es das nachstehende Bild zeigt. Von dort lassen sich die Quellcodes auch leichter herauskopieren:



22.12.2022

Für den absoluten Anfänger ist diese Konfiguration geeignet, um schnell erste Erfolge mit Asterisk vorweisen zu können, um ein paar Telefone über das eigene LAN (das lokale Netzwerk) als Haustelefonanlage zu verbinden. In einem weiteren Schritt lassen sich die Telefone über das Internet weltweit erreichen. Unser Ziel ist ein eigenes Telefonsystem, das uns von kommerziellen Anbietern unabhängig macht.

Über das Internet "nach draußen" kann man mit dieser Konfiguration noch nicht telefonieren. Dafür spart man sich die Portweiterleitung in seinem Router und man muss sich nicht um Schutzmechanismen wie Fail2Ban kümmern, um sich vor SIP-Scannern zu schützen, da ein Zugang zum Internet gesperrt ist. Die Konfiguration als Haustelefonanlage würde auch ohne Internet funktionieren.

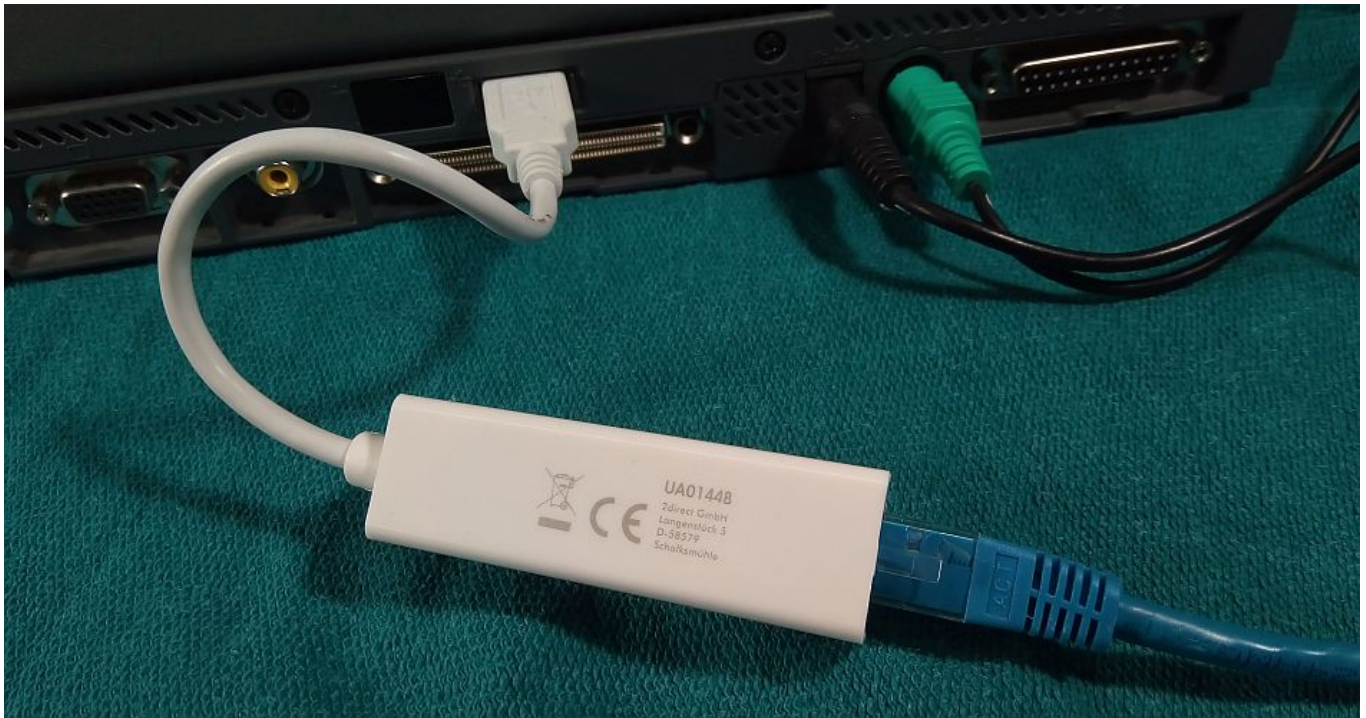
Der Rechner für Asterisk: Was man braucht, ist ein Linux-Rechner, auf dem das Asterisk läuft. Für den Anfang muss es nicht unbedingt ein Raspberry Pi sein. Diese Konfiguration habe ich auf einem alten Notebook Baujahr 1999 getestet, auf dem ich ein altes Ubuntu installiert habe. Der Rechner, auf dem sich Asterisk läuft, sollte eine statische IP erhalten. Für den Anfang funktioniert es auch für eine gewisse Zeit mit

der üblichen dynamischen IP. Der Asterisk-Telefonanlage läuft übrigens mit dem SIP-Protokoll, das die größte Verbreitung hat.



Auf diesem alten Notebook Baujahr 1999 läuft ein altes Lubuntu als Linux-Betriebssystem für den Asterisk-Server.

Den uralten Asterisk-Rechner mit dem LAN verbinden: Am einfachsten geht es über WLAN. Allerdings ist WLAN oft gestört, was zu rätselhaften Fehlern führen kann. Besser ist es den Rechner über ein Netzkabel mit seinem LAN zu verbinden. Mein altes Notebook hatte noch keine Buchse für das Netzkabel. Ich kaufte mir für ein paar Euro einen kleinen Adapter USB auf Ethernet. Dieser macht aus der USB-Buchse eine Buchse für das Netzkabel. Der Adapter wurde ohne Installation sofort von Linux erkannt.



Dieser Adapter macht aus der USB-Buchse eine Netzwerk-Buchse. Der Adapter wurde automatisch unter Windows oder Linux erkannt. Eine Treiber-Installation ist nicht notwendig.

Installation von Asterisk auf Linux am Beispiel von Ubuntu: Die Installation von Asterisk war auf meinem Ubuntu denkbar einfach. Auf dem Raspberry und vielen anderen Linux-Distributionen geht es ebenso. Wer gehen auf der Kommandozeilenebene (LXTerminal auf der grafischen Ubuntu-Oberfläche aufrufen) auf die Root-Ebene mit

Enlighter: Generic Highlighting

```
sudo su
```

und melden uns mit unserem Passwort an, das abgefragt wird.

Dann können wir Asterisk installieren:

Enlighter: Generic Highlighting

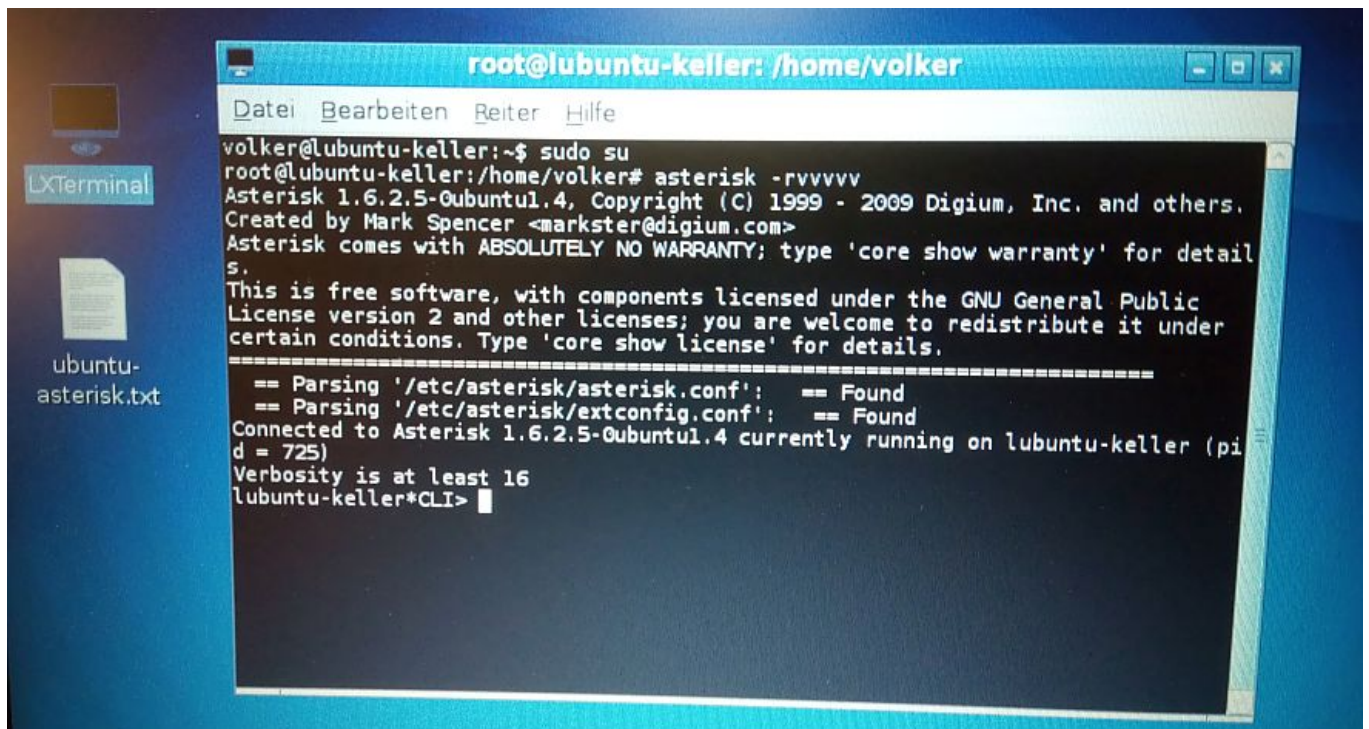
```
sudo apt-get update  
sudo apt-get install asterisk
```

Das war es schon. Asterisk läuft jetzt immer automatisch im Hintergrund, sobald wir den Rechner hochfahren. Um auf die Asterisk-Konsole zu gelangen, geben wir folgendes ein:

Enlighter: Generic Highlighting

```
asterisk -rvvvvvv
```

Die Anzahl der v gibt für die Stufe der Fehlermeldungen an. Wir sollten mindestens drei v eintippen. Nach dem Abschicken des Kommandos müsste sich die Asterisk-Konsole mit einem Text melden, der auch die Asterisk-Versions-Nummer verrät. Bei meinem alten Lubuntu war es die uralte Version 1.6, die für eine alte Haustelefonanlage vollkommen ausreichend ist.



Aufruf von Asterisk in der Konsole (LXTerminal). Installiert ist die Asteriskversion 1.6.2.5. Für die Haustelefonanlage reicht diese uralte Version aus dem Jahr 2009.

Immer wenn wir unsere Konfiguration geändert haben, müssen wir sie in der Asterisk-Konsole mit

Enlighter: Generic Highlighting

```
reload
```

aktualisieren.

Mit

Enlighter: Generic Highlighting

```
sip show peers
```

sehen wir eine Liste der konfigurierten Telefonanschlüsse und ob sie angemeldet sind.

Mit

Enlighter: Generic Highlighting

```
sip show channels
```

werden die laufenden Telefongespräche angezeigt. Da wir noch keine Telefonanschlüsse konfiguriert haben, werden die Rückmeldungen entsprechend sein.

Die Asterisk-Konsole verlassen wir mit

Enlighter: Generic Highlighting

```
exit
```

Wohlgemerkt ist dann Asterisk nicht beendet. Es läuft noch im Hintergrund weiter. Es gibt noch viel mehr zu wissen. Das ist aber alles, was wir als Anfänger wissen müssen.

Was ist die IP des Asterisk-Rechners: Diese IP benötigen wir für die Konfiguration von Asterisk. Wir bringen sie mit

Enlighter: Generic Highlighting

```
ifconfig
```

in Erfahrung.


```

volker@ubuntu-keller:~$ ifconfig
eth0      Link encap:Ethernet  Hardware Adresse 00:e0:4c:19:71:36
          inet Adresse:192.168.1.42  Bcast:192.168.1.255  Maske:255.255.255.0
          inet6-Adresse: fe80::2e0:4cff:fe19:7136/64  Gültigkeitsbereich:Verbindu

          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metrik:1
          RX packets:242245 errors:0 dropped:0 overruns:0 frame:0
          TX packets:125935 errors:0 dropped:0 overruns:0 carrier:0
          Kollisionen:0 Sendewarteschlangenlänge:1000
          RX bytes:42248851 (42.2 MB)  TX bytes:25958935 (25.9 MB)

          Link encap:Lokale Schleife
          inet Adresse:127.0.0.1  Maske:255.0.0.0
          inet6-Adresse: ::1/128  Gültigkeitsbereich:Maschine
          UP LOOPBACK RUNNING  MTU:16436  Metrik:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          Kollisionen:0 Sendewarteschlangenlänge:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

volker@ubuntu-keller:~$

```

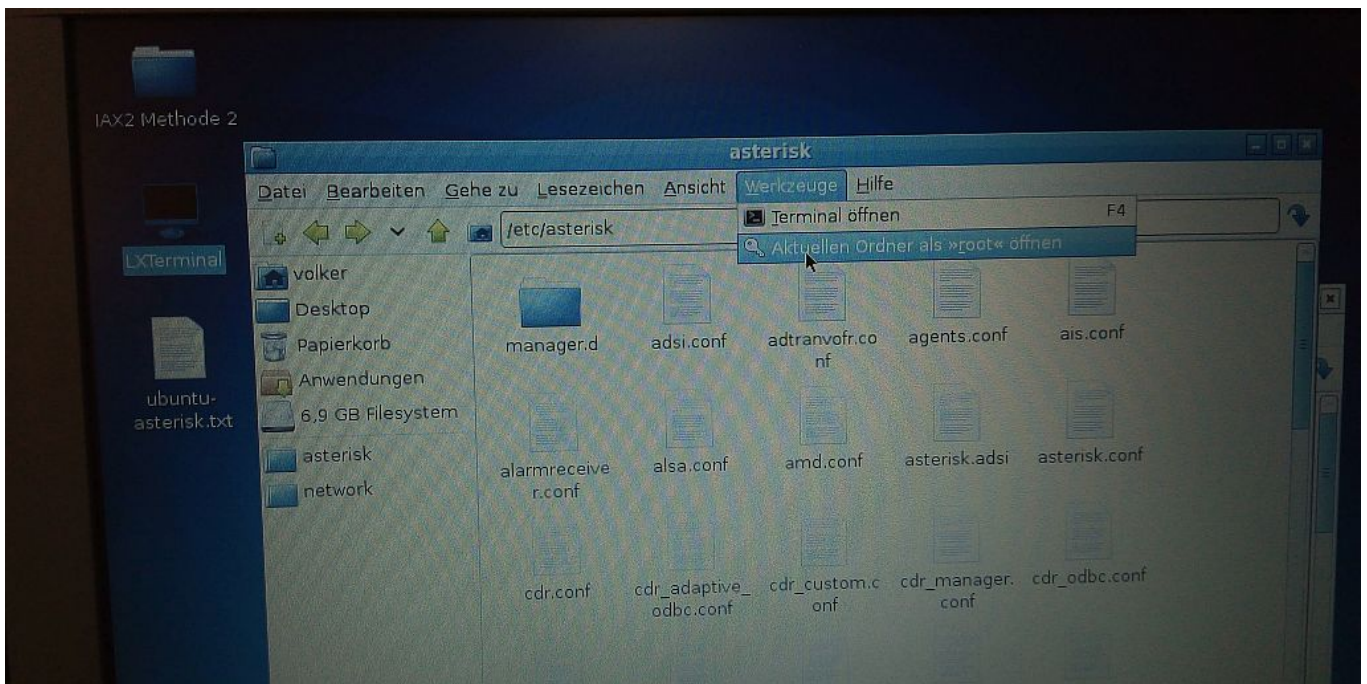
Der Befehl "ifconfig" offenbart die IP 192.168.1.42 des Asterisk-Servers.

Wenn wir auf Ubuntu eine statische IP einrichten wollen, finden wir z.B. [hier](#) eine Anleitung.

Die Konfigurationsdateien: Für unseren minimalistischen Asterisk-Server müssen wir nur drei Asterisk-Konfigurations-Dateien konfigurieren:

sip.conf
extensions.conf
rtp.conf

Die Dateien befinden sich unter /etc/asterisk auf den meisten Asterisk-Versionen. Da wir die Dateien verändern müssen, benötigen sie entsprechende Zugriffsrechte. Bei Ubuntu lässt sich der aktuelle Ordner als "root" öffnen, wie es das nachfolgende Foto zeigt:



Hier befinden sich die Konfigurationsdateien von Asterisk.

Dank der grafischen Benutzeroberfläche gestaltet sich das Editieren der Konfigurationsdateien einfach. Die vorgefertigten Konfigurationsdateien lassen sich auch per SFTP oder noch einfacher per USB-Stick übertragen.

Die sip.conf: Das nachfolgende Beispiel ist für vier Telefone von 9001 bis 9004 ausgelegt. Die IP des Asterisk-Rechners lautet 192.168.1.42 und ist unter localnet eingetragen.

Enlighter: Generic Highlighting

```
; Eine einfache sip.conf nur für die
; Kommunikation innerhalb des eigenen LAN
; mit 4 Sip-Accounts mit den Telefonnummern 9001 bis 9004
; getestet auf Asterisk 1.6.2
;
; Der Registrar für diese Konfiguration lautet
; 192.168.1.42:5075
; Wenn man etwas anderes haben möchte, dann sind die Einträge unter port, bindport und localnet
; entsprechend anzupassen. Der Eintrag in localnet entspricht der IP des Gerätes, auf dem
; der Asterisk-Server läuft. Mit "ifconfig" erfährst du die IP des Linux-Rechners.
```

```
[general]
```

```
; ##### Hier anpassen! #####
port=5075          ; üblich 5060, aber schon von FritzBox belegt.
bindport=5075      ; üblich 5060, aber schon von FritzBox belegt.
; #####
context=telefone ; Context in der extensions.conf muss entsprechend sein
bindaddr=0.0.0.0
```

```
; die Audio-Codecs
disallow=all
allow=alaw
allow=ulaw
allow=gsm
```

```
allow=g722

srvlookup=yes

; ##### Hier anpassen! #####
localnet=192.168.1.42/255.255.255.0 ; Das ist die statische lokale IP des Rechners, auf dem sich
Asterisk befindet
; #####
localnet=192.168.0.0/255.255.0.0
localnet=10.0.0.0/255.0.0.0

nat=no ; nat nicht notwendig, da wir nicht über den Router gehen
insecure=invite
qualify=yes
tos_audio=ef
registerattempts=0
registertimeout=1

; Jitter-Buffer (nicht unbedingt notwendig
; http://www.asteriskdocs.org/en/2nd_Edition/asterisk-book-html-chunk/asterisk-APP-A-SECT-2.html
jbenable=yes
jbforce=yes
jbimpl=adaptive
jbmaxsize=200
jbresyncthreshold=1000

; Vier Telefonnummern 9001 bis 9004

[t1](!)
context=telefone
host=dynamic
canreinvite=no
dtmfmode=rfc2833
type=friend
port=5075

[9001](t1)
username=9001
secret=blabla
callerid = "Robert Redford" <9001>

[9002](t1)
username=9002
secret=faferedd
callerid = "Donald Duck" <9002>

[9003](t1)
username=9003
secret=ereresss
callerid = "Heinrich Heine" <9003>

[9004](t1)
username=9004
secret=xxxaaxxx
callerid = "Angela Merkel" <9004>
```

Die extensions.conf: In ihr stehen die Wahlregeln. Zusätzlich ist noch die Testnummer 222 für den Echotest eingebaut.

Enlighter: Generic Highlighting


```

; extensions.conf

[general]

[telefone]
exten => _9XXX,1,Dial(SIP/${EXTEN},60,rtT)

; ECHOTEST (HALL, ZEITVERZOEGERUNG) Wähle die 222, um den Echotest zu starten.
exten => 222,1,NoOp()
same => n,Answer() ; Abheben
same => n,Playback(demo-echotest) ; Ansage
same => n,Echo ; Echotest
same => n,Playback(demo-echodone) ; Bestaetigung
same => n,Hangup() ; Auflegen

```

Die rtp.conf: Sie sorgt dafür, dass die Sprachübertragung funktioniert. Ich gehe davon aus, dass es auch mit der bereits vorhandenen rtp.conf funktionieren würde.

Enlighter: Generic Highlighting

```

; rtp.conf
; RTP Configuration
;
[general]
;
; RTP start and RTP end configure start and end addresses
;
; Defaults are rtpstart=5000 and rtpend=31000
;
rtpstart=11000
rtpend=11200
;
; Whether to enable or disable UDP checksums on RTP traffic
;
;rtpchecksums=no
;
; The amount of time a DTMF digit with no 'end' marker should be
; allowed to continue (in 'samples', 1/8000 of a second)
;
; dtmftimeout=3000
; rtcpinterval = 5000 ; Milliseconds between rtcp reports
; (min 500, max 60000, default 5000)
;
; Enable strict RTP protection. This will drop RTP packets that
; do not come from the source of the RTP stream. This option is
; disabled by default.
; strictrtp=yes

rtpchecksums=yes
strictrtp=yes

```

Die SIP-Telefone anmelden: Man braucht sich für die Experimente auch keine IP-Telefone anschaffen. Entweder tragen wir die SIP-Accounts unserer Asterisk-Haustelefone in unsere FritzBox ein oder wir verwenden unsere Smartphones, auf dem wir SIP-Softphones wie Siptnetic, SessionTalk oder Linphone aus dem Google

Playstore installieren. Für Windows gibt es das ebenfalls kostenlose Softphone PhonerLite, mit dem ich sehr gute Erfahrungen gemacht habe.

Für die Nummer 9001 gelten zum Beispiel folgende Konfigurationsangaben:


Benutzername/User/Telefonnummer: 9001

Registrar: 192.168.1.42:5075

Passwort: blabla

Erklärung zum Registrar: Vor dem Doppelpunkt steht die IP-Adresse des Asterisk-Rechners. Hinter dem Doppelpunkt der Port, den wir für das SIP-Protokoll festgelegt haben.

Wie wir die Daten der SIP-Accounts in die FritzBox oder die Softphones eintragen, ist nachfolgend beschrieben:

 <p>Die meisten Fritz!Boxen beherrschen Telefonie mit VoIP</p>	<p><u>Einen neuen Telefonanschluss in eine Fritz!Box eintragen - 7.11.2022:</u> Eine Schritt-für-Schritt-Anleitung wie man weitere SIP-Accounts in seine FritzBox einträgt. Die Account-Daten erhält man von seinem Telefonanbieter. - weiter -</p>
---	--

 <p>Das SIP-Softphone SessionTalk auf einem Smartphone.</p>	<p><u>SessionTalk, ein praktisches SIP-Softphone für Android und das Iphone - 1.05.2022:</u> SessionTalk ist ein kompaktes und praktisches SIP-Softphone für Android und das Iphone. Die Einrichtung eines SIP-Accounts hat sich als einfach gestaltet. Es ist selbstverständlich kostenlos und kommt ohne Werbung aus. Zum finden ist es zum Beispiel im Google Playstore. SessionTalk ist besonders für Anfänger empfehlenswert. - weiter -</p>
--	--

Ob wir ein Telefon erfolgreich angemeldet haben, lässt sich in der Asterisk-Konsole mit

Enlighter: Generic Highlighting

```
sip show peers
```

feststellen.

```
lubuntu-keller*CLI> sip show peers
Name/username      Host                Dyn Nat ACL Port      Status
9001/9001           192.168.1.103       D   D   5060     OK (15 ms)
9002/9002           (Unspecified)       D   D   0        UNKNOWN
9003/9003           (Unspecified)       D   D   5060     UNKNOWN
9004/9004           (Unspecified)       D   D   5060     UNKNOWN
4 sip peers [Monitored: 1 online, 3 offline Unmonitored: 0 online, 0 offline]
lubuntu-keller*CLI>
```

Die Nummer 9001 ist auf dem Gerät mit der IP 192.168.1.103 erfolgreich angemeldet.

Unsere Telefonanlage soll auch weltweit über das Internet erreichbar sein: Das wäre dann der nächste Schritt, der hier beschrieben ist:



Für die ersten Schritte mit Asterisk sind die drei wichtigsten Konfigurationsdateien bereits einsatzbereit vorkonfiguriert.

Einfache Asterisk-Konfiguration für zwei SIP-

Telefone - 6.11.2022: Für den Einsteiger stellt dieser Artikel eine einfache Asterisk-Konfiguration vor, die es ermöglicht mit zwei Telefonen mit dem SIP-Protokoll über das Internet zu telefonieren. Wenn dies klappt, haben wir die größte Hürde für den Einstieg in Asterisk genommen. – [weiter](#) –

Weiterführende Informationen: Das war nur der Anfang, wenn wir zwischen zwei oder mehr Telefonen Gespräche führen können. Es geht aber noch viel viel weiter. Wir können ein eigenes weltweit erreichbares Telefonnetz mit Hunderten Teilnehmern aufbauen und unser eigenes Telefonnetz mit anderen Netzen verknüpfen:



Auf einem Raspberry Pi lässt sich ein kompletter Telefonserver betreiben, der seinen Teilnehmern über Landesgrenzen hinweg kostenloses Telefonieren von unterwegs oder daheim ermöglicht.

[Asterisk-Telefonserver auf einem Raspberry Pi -](#)

[Installation und Konfiguration - 2.11.2022:](#)

Diese Seite richtet sich an jene, welche einen Asterisk-Telefon-Server auf einem Raspberry Pi betreiben möchten und später ein kleines Netzwerk aus Asterisk-Servern planen, um ein eigenständiges Telefonnetz aufzubauen. Los geht es mit der Installation von Raspbian und Asterisk auf einem Raspberry Pi und dann nach Lust und Laune immer tiefer in die Programmierung von Asterisk.

Selbstverständlich muss es nicht unbedingt ein Raspberry Pi sein. Andere Linux-Rechner können selbstverständlich auch als Plattform für Asterisk dienen. Allerdings sollte ein Telefon-Server im Dauerbetrieb laufen und dafür eignet sich ein Raspberry Pi, weil er sehr wenig Strom benötigt und Hunderte von Telefon-Teilnehmern bedienen kann. -

[weiter](#) -

