Nextcloud Talk mit eigenem TURN-Server (coturn) » DecaTec

Jan

21. November 2018Home-Server,

Für Nextcloud ist schon seit einiger Zeit eine Erweiterung als App verfügbar, mit der Chats und (Video-)Telefonate über die eigene Cloud geführt werden können: Nextcloud Talk.

Im Normalfall muss man dafür einfach die App im Nextcloud App Store herunterladen und kann sofort loslegen, mit anderen Nutzern der Cloud zu kommunizieren. Dank der Verfügbarkeit von mobilen Apps für Android und auch iOS kann diese Lösung auch als Alternative zu den bekannten Kommunikation-Apps (wie z.B. WhatsApp) genutzt werden. Das Hauptargument für Nextcloud Talk ist dabei sicherlich, dass Chats und (Video-)Telefonate komplett verschlüsselt über die eigene Nextcloud-Instanz ablaufen und so vor den neugierigen Blicken von Konzernen geschützt sind.

Dennoch treten bei Nextcloud Talk oftmals Probleme auf, wenn sich die genutzten Endgeräte nicht im selben Netzwerk befinden. Hier können dann keine (Video-)Telefonate geführt werden, da sich die Geräte einfach nicht finden können.

Damit über Nextcloud Talk jederzeit problemlos kommuniziert werden kann, ist ein eigener TURN-Server notwendig. Mit coturn ist ein solcher auch unter Ubuntu verfügbar. Dieser Artikel beschreibt daher die Installation und Konfiguration von coturn, um optimal mit Nextcloud Talk zusammen zu arbeiten.

Als Basis dient wie immer der Artikel Nextcloud auf Ubuntu Server 18.04 LTS mit nginx, MariaDB, PHP, Let's Encrypt, Redis und Fail2ban.

Wichtig: Auch wenn ein eigener TURN-Server bei Nextcloud Talk zum Einsatz kommt, sind dennoch Videokonferenzen mit maximal **4 bis 6 Teilnehmern** möglich. Für mehr Teilnehmer ist dann ein externer „Signaling Server" notwendig, den man allerdings nur in Verbindung mit einer Nextcloud Enterprise Subscription nutzen kann.
Wenn an die Videokonferenz-Lösung höhere Ansprüche gestellt werden, könnte Jitsi Meet eine geeignetere Lösung sein (siehe Artikel Jitsi Meet: Videokonferenz-System unter Ubuntu Server mit nginx)

Update-Historie (letztes Update: 12.04.2020)

- **12.04.2020:**
  - Hinweis zu den Einschränkungen von Nextcloud Talk hinzugefügt.
- **03.02.2020:**
  - Cipher Suite *DHE-RSA-AES256-GCM-SHA384* entfernt.

Probleme mit Nextcloud Talk mit eigenem STUN/TURN-Server lösen

Nextcloud Talk basiert auf WebRTC. Die bei der Kommunikation beteiligten Endgeräte bauen dabei eine direkte Peer-to-Peer-Verbindung auf.

Im **gleichen Netzwerk (LAN)** ist dies kein Problem, hier müssen alle Endgeräte lediglich WebRTC unterstützen.

Wenn sich die Geräte in **unterschiedlichen Netzwerken** befinden, müssen die Geräte sowohl ihre interne, also auch ihre externe IP-Adresse kennen. Diese Umsetzung ist die Aufgabe eines STUN-Servers (Session Traversal Utilities for NAT). Für Nextcloud Talk ist ein eigener STUN-Server verfügbar (*stun.nextcloud.com:443*), daher sollte dies auch kein Problem darstellen.

Wenn nun allerdings **Firewalls** mit im Spiel sind, kann der STUN-Server die „Übersetzung“ der Adressen nicht mehr leisten. In diesen Fällen spricht man von einem Symmetric NAT: Hier wird durch die Firewall verhindert, dass von außen (d.h. aus dem Internet) initiierte Verbindungen in das lokale Netzwerk möglich sind. In diesem Szenario ist dann ein TURN-Server (Traversal Using Relays around NAT) notwendig, über den sämtliche Verbindungen geleitet werden.

Diese Thematik ist durchaus komplex, aber vereinfacht kann man sagen, dass man einen TURN-Server benötigt, wenn die Verbindung zwischen Geräten in unterschiedlichen Netzwerken nicht aufgebaut werden kann (z.B. PC im Heimnetzwerk, Smartphone in Mobilfunknetz). Im Heimnetzwerk-Bereich ist dies vermutlich immer der Fall.

Unter Ubuntu ist coturn als quelloffene Implementierung eines STUN/TURN-Servers verfügbar.

Installation und Konfiguration coturn

coturn kann ganz einfach auf dem gleichen System installiert werden, auf dem auch schon Nextcloud läuft.

Zunächst bringen wir das System auf den neusten Stand:

```
apt-get update && apt-get upgrade -V
```

Anschließend kann coturn auch schon installiert werden, da es bereits Teil der Ubuntu-Paketquellen ist:

Nach der Installation muss coturn nun nur noch konfiguriert werden, damit eine optimale Zusammenarbeit mit Nextcloud Talk gewährleistet ist.

Dazu wird coturn erst einmal aktiviert. Dies geschieht in folgender Datei:

Hier muss folgende Zeile eingefügt werden. Diese ist nach der Installation bereits vorhanden, ist allerdings auskommentiert. Daher entfernen wird einfach das Zeichen ‚#' am Anfang der Zeile:

Als nächstes werden die Einstellungen von coturn bearbeitet.
Diese befinden sich in folgender Datei:

```
nano /etc/turnserver.conf
```

Hier müssen die folgenden Werte angepasst werden. Die meisten Werte sind hier bereits vorhanden, allerdings auskommentiert. Auch hier entfernen wir die Raute (‚#') vor der entsprechenden Zeile.

- tls-listening-port=5349
- fingerprint
- lt-cred-mech
- use-auth-secret
- static-auth-secret=<secret>
- realm=meinedomain.de
- total-quota=100
- bps-capacity=0
- stale-nonce=600
- cert=/etc/letsencrypt/live/meinedomain.de/cert.pem
- pkey=/etc/letsencrypt/live/meindomain.de/privkey.pem
- cipher-list="ECDHE-RSA-AES256-GCM-SHA512:DHE-RSA-AES256-GCM-SHA512:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384"
- no-loopback-peers
- no-multicast-peers
- dh-file=/etc/nginx/ssl/dhparams.pem
- no-tlsv1
- no-tlsv1_1
- no-stdout-log

Zu den angegebenen Werten hier noch ein paar Erklärungen:

- `tls-listening-port`

  Hier verwende ich den Standard-Port für TLS-Verbindungen zu coturn (5349). Hier könnte man auch einen anderen Port angeben. Diesen sollte man sich merken, da man diesen nachher in der Firewall freischalten muss.

- `static-auth-secret`

  Dies ist ein Passwort, welches zur Nutzung des TURN-Servers benötigt wird. Dies ist ein Sicherheitsmerkmal, so dass kein Dritter ohne Kenntnis dieses Passwortes den TURN-Server verwenden kann. Dieses Passwort muss man sich nicht merken und wird später nur in den Nextcloud-Einstellungen hinterlegt, daher lässt man sich dies am besten einfach durch folgenden Befehl generieren: *openssl rand -hex 32*

- `cert/pkey/dh-file`

Dies sind die SSL-Zertifikate bzw. Diffie-Hellman-Parameter, die für die verschlüsselte Verbindung über TLS benötigt werden. Hierzu verwenden wir einfach die bereits vorhandenen Zertifikate für die eigene Domain (siehe hier und hier).

- `cipher-list`

  Legt die Cipher-Suite für die TLS-Verbindung zum TURN-Server fest. Hier kann die Cipher-Suite angegeben werden, die auch schon der Webserver nginx verwendet (hier heißt die Variable *ssl_ciphers*, siehe hier).

Nach der Anpassung der Konfiguration von coturn wird das Programm neu gestartet, damit die Änderungen übernommen werden.

Portfreigaben einrichten

Nach der Einrichtung des TURN-Servers muss nun noch eine Portfreigabe eingerichtet werden, damit coturn auch „von außen“ erreichbar ist.

Wichtig ist hier der Port, der bei der Variable *tls-listening-port* in der coturn-Konfiguration angegeben wurde. In diesem Beispiel verwende ich dazu den Standard-Port 5349.

Zunächst muss hier eine entsprechende Portweiterleitung im Router eingerichtet werden. Das genaue Vorgehen unterscheidet sich hier von Router zu Router, daher kann an dieser Stelle keine detaillierte Anleitung erfolgen. Hier sollte man aber alle notwendigen Informationen auf den Hersteller-Seiten finden (so z.B. auf den AVM Hilfeseiten, wenn man eine FritzBox im Einsatz hat). Wichtig ist hier, dass die Freigabe für die beiden Protokolle TCP und UDP angelegt werden. Dazu ist meistens das Anlegen von zwei Freigaben notwendig, hier am Beispiel einer FritzBox:

Portfreigaben für cotun (FritzBox)

Wenn darüber hinaus auch noch eine Firewall auf dem System installiert ist, auf dem coturn eingerichtet wurde, so muss auch hier eine Freigabe erfolgen. Auf vielen Systemen wird wohl ufw (**u**ncomplicated **f**ire**w**all) eingerichtet sein. Hier erfolgt die Freigabe dann über folgende Befehle:

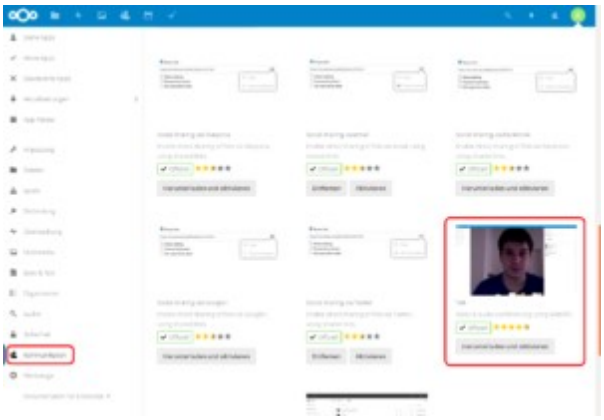
```
ufw allow 5349/tcp
ufw allow 5349/udp
```

Nextcloud Talk mit eigenem STUN/TURN-Server

Nun kann Nextcloud Talk mit dem eigenen STUN bzw. TURN-Server betrieben werden.

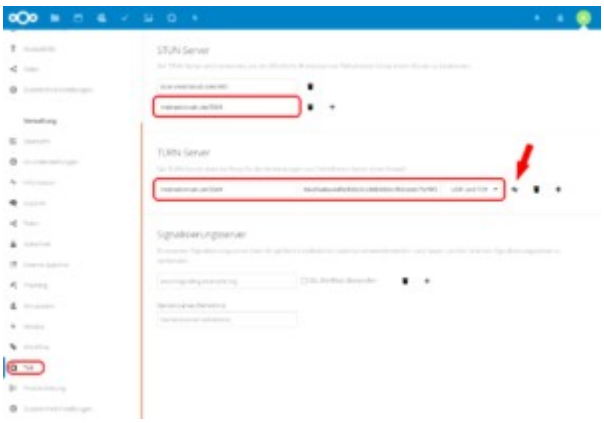
Zunächst muss – falls noch nicht geschehen – die App Talk aus dem Nextcloud App Store heruntergeladen werden. Diese befindet sich in der Kategorie

Kommunikation.



Talk im Nextcloud App Store

Nach der Installation der App muss diese noch konfiguriert werden, damit diese coturn als STUN bzw. TURN-Server nutzt. Die dazugehörigen Optionen findet man in den Admin-Einstellungen von Nextcloud unter dem Punkt *Talk*.



Nextcloud Talk: Einstellungen für eigenen STUN/TURN-Server

Zunächst fügen wir mit der Plus-Schaltfläche einen **zusätzlichen** STUN-Server hinzu:

meinedomain.de:5349

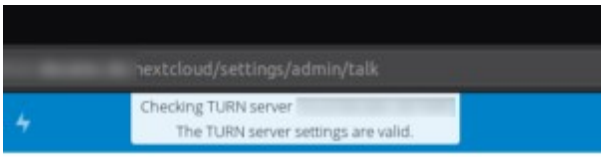
Die Einstellungen für den TURN-Server sehen dann folgendermaßen aus:

- *URI*: *meinedomain.de:5349*: Der Port wurde zuvor in der coturn-Konfiguration (*tls-listening-port*) hinterlegt.
- *Gemeinsames Geheimnis*: Hier muss das Passwort angegeben werden, welches bei der Konfiguration von coturn als Variable *static-auth-secret* hinterlegt wurde.
- *UDP und TCP*: Nextcloud Talk soll hier für maximale Kompatibilität sowohl TCP- als auch UDP-Verbindungen nutzen.

Unter *Signalisierungsserver* sind keine Eingaben notwendig.

Die Konfiguration kann nun ganz einfach mit der entsprechenden Schaltfläche getestet werden (siehe Pfeil oberes Bild). Hier sollte dann nach einem kurzen

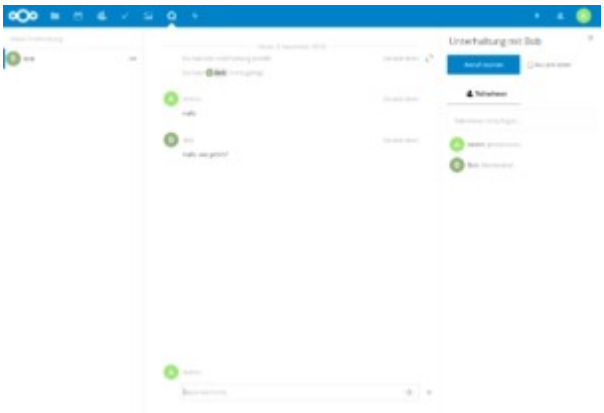
Augenblick auch eine Erfolgsmeldung erscheinen:



TURN-Server ist richtig konfiguriert

Chats und (Video-)Telefonie über die eigene Nextcloud

Ab sofort können über Nextcloud Talk Chats, Telefonate und auch Video-Telefonate geführt werden. Dies ist auf den verschiedensten Endgeräten möglich: Zum einen kann Nextcloud Talk ganz einfach im Browser verwendet werden. Dazu in der oberen Menüleiste einfach auf das Talk-Symbol klicken. Nun noch den passenden (Nextcloud-)Kontakt auswählen und schon kann es losgehen.



Nextcloud Talk in Aktion

Richtig interessant wird die Sache allerdings mit mobilen Geräten, wie z.B. Smartphones. Hier stehen App sowohl für iOS, als auch für Android bereit:

Der **Funktionstest** des eigenen TURN-Servers erfolgt am besten mit einem Smartphone: Wenn dieses nicht mit dem Heimnetzwerk, sondern mit dem Mobilfunknetz verbunden ist, sollten sowohl Chats, auch als (Video-)Telefonate mit einem Gerät im Heimnetzwerk problemlos möglich sein.

Troubleshooting

Falls Probleme mit Nextcloud Talk auftreten, sollte zunächst einmal die grundsätzliche Konfiguration durch die entsprechende Schaltfläche in den Admin-Einstellungen zu Talk überprüft werden (siehe Screenshot weiter oben). Anschließend kann es auch sinnvoll sein, die **Nextcloud-Logs** zu überprüfen. Diese findet man in der Admin-Oberfläche von Nextcloud unter dem Punkt *Protokollierung*.

Wenn im Nextcloud-Log keine Einträge vorhanden sind, die sich mit Talk in Verbindung bringen lassen, sollte die Funktion von coturn überprüft werden. Die **Log-Dateien von coturn** befinden sich im Verzeichnis */var/log*. Die gesuchten Log-Dateien beginnen alle mit *turn_* und beinhalten eine Datumsangabe im Dateinamen, z.B. *turn_1352_2018-11-08.log*. In diesen Log-Dateien sollte dann zumindest ein Hinweis zu finden sein, warum es zu Problemen gekommen ist.

Fazit

Im Heim-Bereich funktioniert Nextcloud Talk leider nicht immer out-of-the-box – mit dem Aktivieren der App ist es hier meistens nicht getan. Allerdings kann mit coturn recht einfach ein eigener STUN/TURN-Server in Betrieb genommen werden. Der entsprechende Konfigurations-Aufwand hält sich hierfür auch in Grenzen.

Mit Hilfe des eigenen TURN-Servers ist es nun aber möglich, über die eigene Cloud Chats und (Video-)Telefonate zu führen. Dieses Konzept ist im Vergleich zu den bekannten Apps (wie z.B. WhatsApp) besonders interessant, da die Kommunikation ausschließlich verschlüsselt über die persönliche Cloud abläuft, ohne dass hier Konzerne wie Google oder Facebook „mitlesen“ können.

Weiterführende Artikel

- Nextcloud auf Ubuntu Server 18.04 LTS mit nginx, MariaDB, PHP, Let’s

Encrypt, Redis und Fail2ban
- Ubuntu Server 18.04 LTS als Hyper-V Gastsystem installieren und optimal einrichten
- Jitsi Meet: Videokonferenz-System unter Ubuntu Server mit nginx

Links

- Nextcloud Homepage (englisch)
- Übersichtsseite Nextcloud Talk (englisch)
- Nextcloud Talk im Nextcloud App Store
- WebRTC (Wikipedia)
- Peer-to-Peer (Wikipedia)
- Netzwerkadressübersetzung (Wikipedia)
- Session Traversal Utilities for NAT (Wikipedia)
- Traversal Using Relays around NAT (Wikipedia, englisch)
- coturn (GitHub)

Cloud, coturn, Home-Server, Linux, Nextcloud Talk, TLS, TURN, Ubuntu, Videokonferenz