

VoIP

Pojęcia

- Voice over Internet Protocol (VoIP) to technologia pozwalająca na prowadzenie rozmów wykorzystując infrastrukturę IP. Ponadto, istnieją rozwiązania pozwalające na zestawianie połączeń z “normalną” telefonią.
- PBX - Private Branch Exchange - prywatna telefoniczna centrala abonencka. Zarządza wewnętrzną siecią telefoniczną np. Firmy.
- Telefony VoIP - mogą być sprzętowe lub programowe (ang. softphone)
- SIP (Session Initiation Protocol) - zestawia połączenie, informuje o zakończeniu połączenia
 - SDP (Session Definition Protocol) - załącznik zawierający “manifest” parametrów
- RTP (Realtime Transport Protocol) - realizuje przesył multimedialny.
- SRTP (Secure RTP) - szyfruje przesyłane multimedia.

SIP

Protokół podzielony na żądania i odpowiedzi (jak HTTP).

Żądania: Invite, Ack, Register, Bye, Cancel, Options, PRACK,

Kody odpowiedzi:

- 1xx = informacyjne (trying, ringing),
- 2xx = sukces (ok),
- 3xx = przekierowanie (moved permanently),
- 4xx = request failure (unauthorized),
- 5xx = server error (Server Internal Error),
- 6xx = global failure (Busy Everywhere)

Asterisk

Darmowy system typu PBX.

Interesujące pliki (/etc/asterisk/):

- sip.conf (definicja użytkownika - telefonu, numeru exten itp.)
- extensions.conf (dial plan, definicja akcji np. Music on hold, menu itp).

Dial plan (logika odbierania połączeń): określa jakie akcje zostaną podjęte po zestawieniu połączenia.

AGI: Application Gateway Interface, umożliwia stworzenie aplikacji w jednym z wielu kompatybilnych języków programowania, która może być uruchomiona przez serwer asterisk.

Konfiguracja SIP

Otwórz plik sip.conf. Usuń wszystkie linijki zawierające wyłącznie komentarz i puste linie.

Dla kontekstu [general] dodaj (pod srvlookup=yes) linijkę qualify=yes

Dodaj (na końcu pliku) nowy wpis dla każdego użytkownika:

```
[nazwa] ; nazwa użytkownika, unikalna
    type=friend
    context=phones
    allow=ulaw,alaw
    secret=12345678 ; hasło, można zmienić, są inne możliwości ustawienia hasła
    host=dynamic ; nie znamy adresu IP
```

W konsoli CLI asterisk (sudo asterisk -rvvv):

```
sip show peers
sip reload
sip show peers
```

Podłączenie klienta

Displayname/account/username: to co podaliśmy jako nazwę w sip.conf

Password: secret z sip.conf

Registrar/domain: ip pod którym jest dostępny serwer asterisk.

Następny krok: możliwość zestawienia połączenia telefonicznego z innym klientem.

Dial plan

Organizacja za pomocą kontekstów (~przestrzeń nazw). Konteksty są od siebie niezależne, rzeczy zdefiniowane w jednym kontekście nie są widoczne z innych kontekstów.

Końcówki (ang. extensions) to identyfikatory użytkowników. Nazwy użytkowników są mapowane na końcówki. Może wystąpić sytuacja gdzie końcówka występuje w dwóch kontekstach.

Dialplan - po wybraniu numeru (klient dzwoni na numer), dialplan szuka, w kontekście właściwym dla dzwoniącego użytkownika, końcówki równoważnej wybranemu numerowi i wykonuje akcje powiązane z tą końcówką.

Dialplan c.d.

Wyczyść plik extensions.conf

Dodaj treść:

```
[phones] ; kontekst
exten => 100,1,NoOp(Raz)
exten => 100,2,NoOp(Dwa)
exten => 100,3,Hangup()
```

Przejdź do konsoli CLI (sudo asterisk -rvvv):

```
dialplan reload
```

Zadzwoń na numer 100, zobacz co zostało wypisane w konsoli CLI.

Lista aplikacji (w CLI):

```
core show applications
```

```
core show applications like/describing X
```

Opis aplikacji (w CLI): core show application X

zwróć uwagę na sekcję syntax dla opisu dial

Dialplan c.d.

Do pliku `extensions.conf` dodaj linijkę wykonującą aplikację `dial`, która zadzwoni do jednego z użytkowników. Przetestuj, dzwoniąc z drugiego użytkownika. Pamiętaj o przeładowaniu konfiguracji (`dialplan reload`).

Uproszczenia w pliku `extensions.conf`:

```
exten => 100,1,[aplikacja]
same => n,[aplikacja] ; same = ta sama końcówka co wyżej (tu: 100)
same => n,[aplikacja] ; n = kolejna liczba (rozpatrywane od góry do dołu)
```

W pliku `extensions.conf` dodaj wpisy dla drugiego użytkownika (możliwe dzwonienie `u1 -> u2` i `u2 -> u1`)

Dialplan c.d.

Dopasowanie (expression) - zaczyna się od _

Np.:

_XXXX - dowolne 4 cyfry

_X. - dowolna cyfra dowolną liczbę razy

Zmienne: wyświetlenie zawartości zmiennej - przez aplikację NoOp (podać jako argument). Np. NoOp(\${EXTEN})

AGI

Zainstaluj bibliotekę pyst2 na koncie root, korzystając z pip.

Stwórz skrypt w pythonie, nadaj uprawnienia do wykonywania. W linii shebang wpisz `#!/usr/bin/python2.7`

```
import sys
from asterisk.agi import *

agi = AGI()
agi.verbose("python agi started") # wypisywanie komunikatów kontrolnych
callerId = agi.env['agi_callerid'] # dostęp do zmiennych
agi.stream_file('vm-extension') # odtwórz dźwięk
result = agi.wait_for_digit(-1) # przechwyć klawisz od dzwoniącego
agi.say_number(result) # odczytaj podaną cyfrę
agi.hangup() # zakończ połączenie
sys.exit() # wyjdź
```

Ćwiczenia

1. Uruchom aplikacje telefonów voip na dwóch maszynach i zestaw połączenie bezpośrednio między nimi
 - a. Stwórz konto, domena: własny adres IP
2. Skonfiguruj telefon IP i wykonaj rozmowę z linphone.
3. Otwórz wireshark, filtruj ruch SIP, RTP i SDP. Wykonaj kilka scenariuszy rozmów (poprawna, odrzucona, przerwana). Jakie kodeki były użyte w sesji?
4. To samo co powyżej, wejdź w menu Telephony->voip calls, flow sequence, zobacz jaki przebieg miała rozmowa
5. Zasymuluj atak podsłuchowy typu man-in-the-middle. Spraw by ruch odbywał się przez dodatkowy host (np. Ustaw go jako bramę domyślną). Na tej maszynie uruchom program wireshark i przechwyć zapis jednej rozmowy. Korzystając z Telephony->voip calls, play streams przesłuchaj treść rozmowy.

Ćwiczenia - asterisk

1. Uruchom asterisk, dodaj użytkownika w konfiguracji (sip.conf). Dokonaj rejestracji wykorzystując softphone lub fizyczny telefon.
2. Zarejestruj kilku użytkowników i wykonaj połączenie między nimi.
3. Spraw by połączenie z wybranym użytkownikiem poprzedzało odtworzenie pewnego dźwięku (dostępne pliki w /usr/share/asterisk/sounds np. tt-monkeys). Wystarczy podać nazwę pliku (bez ścieżki, bez rozszerzenia).
4. Utwórz końcówkę z przekierowaniem (Goto). Opcjonalnie: przekierowanie warunkowe np. GoTolftime()
5. Interaktywne menu wykorzystując wbudowane aplikacje (Answer, Playback, WaitExten).
6. Interaktywne menu wykorzystując AGI (np. python)