International Journal of Electrical and Computer Engineering (IJECE)
Vol. 6, No. 3, June 2016, pp. 1072 ~ 1076
ISSN: 2088-8708, DOI: 10.11591/ijece.v6i3.9843

Special Features of Radio Interception of APCO P25 Messages in Russia

Dmitry Sergeevich Silnov
Department of Information Systems and Technologies, National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Moscow, Russia

Article Info

Article history:

Received Oct 12, 2015
Revised Dec 30, 2015
Accepted Jan 10, 2016

Keyword:

Apco p25
Google Speech
Radio Interception
Yandex Speechkit

ABSTRACT

Many special forces in Russia use APCO P25 encoding while radio talks. Using google speech or yandex speechkit while decoding radio talks can give huge advantage. Search engines can index decoded text or knowledge base can be created.

Corresponding Author:

Dmitry Sergeevich Silnov,
Department of Information Systems and Technologies,
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute),
Kashirskoe sh. 31, Moscow, Russian Federation.
Email: ds@silnov.pro

1. INTRODUCTION

Amateur radio as a hobby appeared a very long time ago, long before we saw the first computers. Until recently, airwaves interception was a complicated task requiring professional skills in radio electronics along with high costs of the equipment capable of receiving waves in wide frequency ranges. Mainly, this costly pastime normally boiled down to listening to the talks between pilots and traffic controllers.

2. TASK RESCRIPTION

When the Chinese USB TV tuners built on the RTL2832 chip appeared, everything became much easier. It became possible to build an inexpensive radio waves scanner without paying a lot. And radio interception is no longer limited by listening to the air talks. Today, portable transceivers are used everywhere by every municipal service, from builders and taxi drivers to all kinds of operational agencies, such as the Ministry of Emergency Situations, the police, and other governmental organizations. The current situation in Russia is that most radio talks are not encrypted at all. Encryption is obligatory only for the radio contacts of Federal Security Service and other special forces that the national security depends on, but those talks only make a small part of the total volume of the radio contacts in the country.

Let's see how we can use a USB TV tuner to make an inexpensive and high-quality radio waves interceptor. A USB tuner with an RTL2832 chip is the hardware that receives the signal. The signal can be both analogue or digital. In case of a digital signal, the decoding is done by the CPU of the computer that the USB receiver is connected with. P25/DMR/Mototrbo/NXDN/ProVoice signal formats can be decoded there [1].

Originally, the receiver specifications show understated values. But actually, the receiver is able to get not only TV signals; the receivable frequency range is very wide and provides solutions for nearly all kinds of tasks for radio amateurs. The only possible limitation of the receivable frequencies is the antenna bundled with the tuner. The antenna supplied by the manufacturer is only intended for high-quality reception of the TV range frequencies, so signals of any other ranges cannot be received with the same efficiency. But this can be easily resolved by replacing the antenna with another one designed for reception of the required frequency range.

Before, one had to use portable transceivers to perform the required tasks and set the frequencies manually to receive signals, and the output signal was analogue, which resulted in low overall quality. Besides, they used special radio equipment that allows listening to both analogue and digital signals, but the cost of this type of equipment is very high, so no ordinary radio amateur could ever afford it. But now the situation has changed: using a cheap USB TV tuner, we can automatically set the desired frequencies very quickly and receive original digital signals with no quality losses.

3. SOLUTION

One of the features of a USB tuner is the ability to build a system for interception of trunk networks where the interception quality becomes a few times better. As trunk networks are intended for talking at different frequencies and one cannot know which frequency is chosen by the base station, you cannot go the classical way to intercept such talks by simply setting one certain frequency.

To listen in to talks over trunk networks, we have to use two USB tuners. Every tuner should be connected with a splitter to a common antenna, or two antennas should be used, with every tuner connected to its own antenna. The Unitrunker app is used to intercept trunk networks. It allows setting each tuner to perform specific tasks. One of the USB tuners is adjusted to work as the signal receiver, and the app uses it to get a list of currently active channels. Using the other tuner, the app directly receives the sound signal from a certain channel. This allows automating the interception of all the channels of a specific trunk network automatic, with no need to set the channels manually.

To make the interception, recording and decoding process automatic in case of digital talks, we have to build a system of various software tools that do not require any permanent adjustments or control. The software tools include: apps to configure the USB tuner and get the signal from it, an app to decode the digital signal, and a special driver app to redirect the output audio signal from the sound card terminal to the audio stream decoder. The Windows OS is not suitable for building such a system, because there are a lot of important tools designed as applications with visual interface and requiring that a human operator should directly configure them. But when the PC reboots, it will be required to restart all the apps and reconfigure them from the very beginning. It is more preferable to build such a system based on FreeBSD.

FreeBSD allows using ready-to-use software tools required for interception of radio waves and saving the results. The system is launched with a single command and requires no human involvement in the operation. This system allows complete automation of the radio waves interception process and also saving the intercepted results. There have been some similar solutions before, but they could only retransmit the sound to various websites without the possibility to listen to the talks history. The system produces read-to-use audio files which allow editing and improving the sound quality, saving the data, and replaying the audio files at any moment. Let's describe each app in the package separately.

Rtl_fm. This utility is designed for operating directly with a USB tuner. It is used to set the operating frequency, the shift, the parameters of the recorded sound, and to adjust some other settings. The output is a sound file in its original condition.

SOX. This is a multitask utility for sound file processing. We input a sound file in its original condition, as taken from the USB tuner, into the utility. The sound is converted depending on the selected parameters. Here we have WAV as the sound file format. Then it is used for further processing of the sound file; it has filters for high and low frequencies and allows removing noises or pauses.

DSD. This utility is used if receiving an encoded signal. The encoded signal is loaded into the utility, and it produces a decoded audio signal that can be listened to online and saved.

The command that launches the system when ready-to-listen WAV files are created. Each file contains individual talks that can be separated with the SOX utility which finds the frames of every talk according to the pauses:

```
rtl_fm -Mfm -f $1 -d 0  -p 22  -s 48k -F 9 -E deemp  | \
sox -v 5.0 -r 48k -e signed-integer -c 1 -b 16 -t raw - -t wav - | \
dsd -u 1 -v99 -i /dev/stdin -o /dev/stdout | \
sox -v 5.0  -e signed-integer -r 8000 -c 1 -b 16 -t raw  -  $curdir/$1_$DT/$1.wav silence 1 0.30 1% 10.3 1%
```

If the system manages to produce audio files with the desired quality, the next feature is recognition of the speech in the audio files and getting the script of the conversations. To do so, we use third-party services, such as Yandex SpeechKit or Google Speech (Figure 1).

Figure 1. Decoding circuit voice calls

When testing the system, we managed to intercept talks at various frequencies. The software tools receiving the data from the USB tuner allow visual demonstration of the currently active frequencies of the radio transmission, with the peaks on the graph showing the talks which are in progress at the moment. When a frequency with talks currently in progress is selected, the signal is being received and the talks can be listened to in the real time mode. The graphs show that the signals have perfect power levels and allow data reception with no losses (Figure 2).

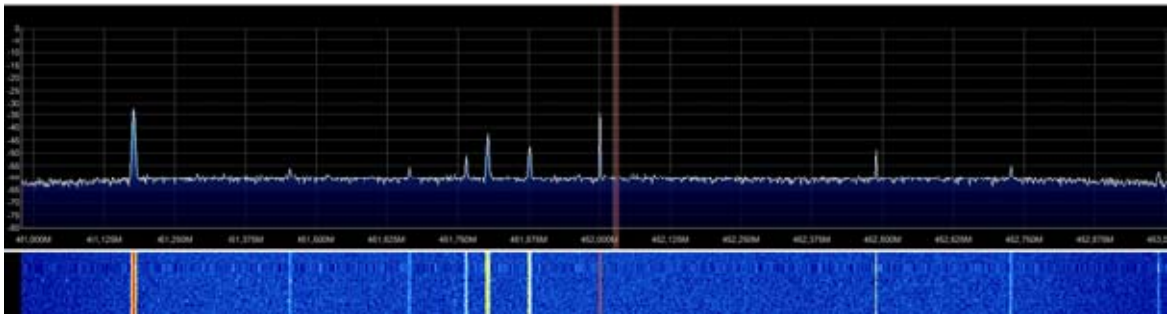

Figure 2. Peaks of encoded signals

The talks at the frequencies around 452 MHz are of greatest interest to detached radio amateurs (Figure 3). This is the frequency used by various divisions of the police, road traffic control department and non-departmental security services. And despite the fact that the talks are transmitted in encrypted digital format (P25 Standard [2]-[4]), the USB tuner allows receiving the signal with high quality without any losses, which gives anyone the possibility to easily read, decode and listen to the talks right away.

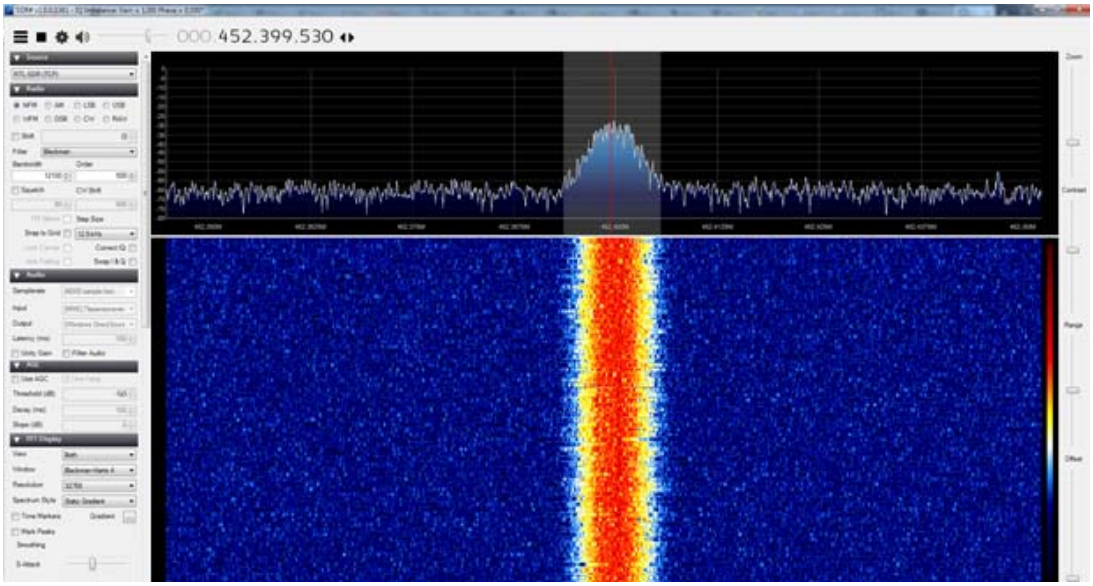

Figure 3. Strength of 452.400 signal

The only problem that may impede the recognition of the talks is the quality of the resulting audio file. For instance, this may occur as a result of indistinct pronunciation, presence of external noise in the waves or low volume. In this case, the voice may not be recognized or recognized improperly [5]. The SOX utility allows removing the faults of the resulting audio file by means of using the built-in filters and editing the volume level.

We also have the possibility to run the service on a remote server computer, with a USB tuner connected to it. Radio waves can be intercepted then on a client PC with internet access from any corner of the globe [6]. In this case, we use the rtl_tcp utility: it should be launched on the server computer as a service, the client gets connected and receives the signal from the USB tuner in its pure form. If necessary, the signal can be decoded and listened to on the client side.

Example of the service launch on a server computer:
rtl_tcp -a 88.151.117.136 -d 1 -p 12345.

Talks examples:
– One ten, what's up?
– Tenth and ninth, three bikes and a parking.
– Zero zero, stir up the parking and look at the tinting.
– Two seven three, Mytishchi, two seven three.
– Two seven three, here.
– Subway bridge, no direction, a black Bentley, a hundred and three digital, head-on crash.
– Got it.
– Kaluga Kaluga, six zero two, one to the house, you called a squad?
– Yeah, we did.
– Proceed to Donskaya, I got the house, there's a unit working in the street, check it out.
– Got it.

4. CONCLUSION

Buying a cheap USB tuner and connecting it to an ordinary personal computer, any radio amateur obtains a powerful instrument for radio waves interception. By using it, anyone can gain access to classified information transmitted over the radio. Of course, the ordinary tuner package only allows receiving the TV signal, but no special knowledge or skills in radio engineering is required to retune the unit. All you need is to install the necessary software, which any advanced PC user is able to do.

The P25 digital standard was supposed to become the barrier preventing from intercepting the radio contacts of the police, road traffic control department and other governmental and private divisions. It does not allow to listen in to the talks in their pure form, but using a USB tuner and special software, you can immediately get those talks decoded. The CPU of an ordinary PC is fully capable of that kind of signal decoding.

Further upgrading of the system brings the possibility to recognize the talks and save them as texts, which allows accumulating the data in as convenient a form as possible. As the data represented in that way can be easily indexed, it could result in a huge knowledge base. This will provide the possibility to easily find and structure the data obtained from the radio talks.

REFERENCES

[1] Kiley P. and Benedett T., "Public safety interoperability with an SCA military radio using the P25 waveform," in *Military Communications Conference, 2007. MILCOM IEEE,* pp.1-8, 2007.
[2] Clark S., *et al.*, "Why (Special Agent) Johnny (Still) Can't Encrypt: A Security Analysis of the APCO Project 25 Two-Way Radio System," in *USENIX Security Symposium*, 2011.
[3] Glass S., *et al.*, "A software-defined radio receiver for APCO Project 25 signals," in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pp. 67-72, 2009.
[4] Cigirkan G., *et al.*, "Efficient and Reliable Multicast of Data in APCO P25 Systems," in *Vehicular Technology Conference (VTC Spring), IEEE 81st*, pp. 1-5, 2015.
[5] J. Joost, "Exploring Possible Vulnerabilities of 868MHz Communication Systems: A Step-By-Step Framework," 2015.
[6] B. Benjamin and L. Shamir, "Assessing the efficacy of benchmarks for automatic speech accent recognition," *Proceedings of the 8th International Conference on Mobile Multimedia Communications. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)*, 2015.

BIOGRAPHY OF AUTHOR

Assosiated Professor at Department of Information Systems and Technologies, National Research Nuclear University MEPhI (Moscow Engineering Physics Institute). Doing researches in the field of information secutiry.