

Take a page from the IT handbook

FOSS applications can be adapted to create custom solutions for broadcast

I consider it my mission to stay on top of tech — always reading, watching and digging. It is my passion to bring custom solutions to broadcast customers.

I frequently am inspired by software, networking and virtualization technologies used across so many industries, tried and true solutions that easily could benefit local radio.

Generally, IT infrastructure already exists in a broadcast facility in the form of routers, firewalls, switches and virtualization stacks from various vendors. It is *imperative* to make sure that this infrastructure is designed correctly, with security and functionality in mind.

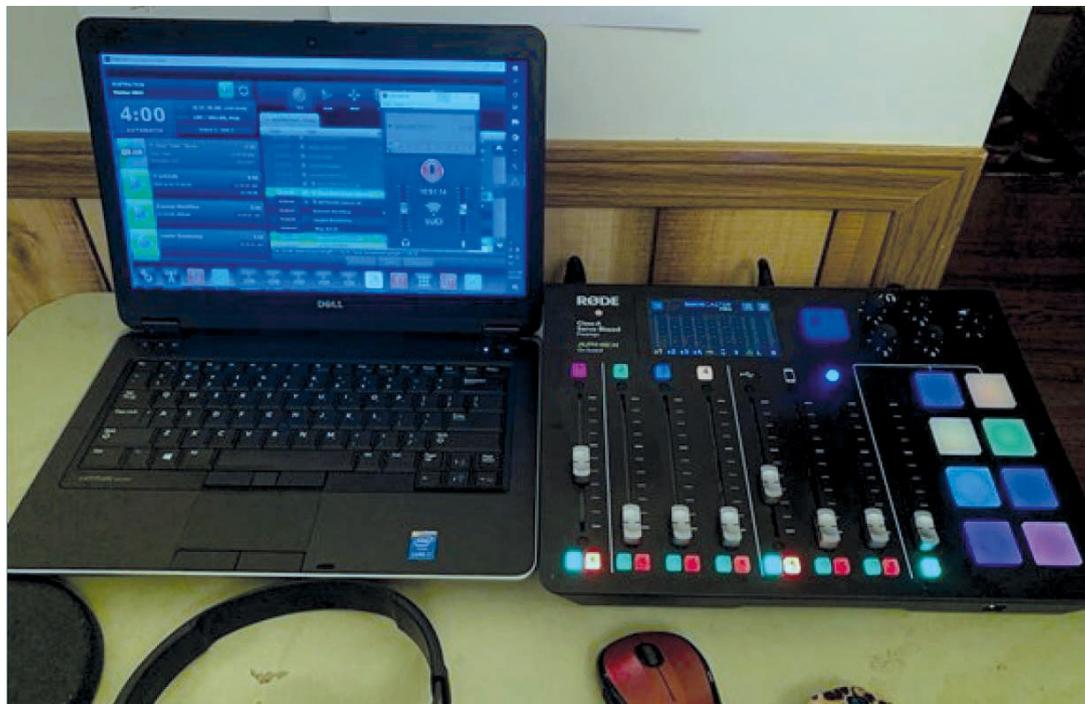
In our industry, most of us have become accustomed to working with less-than-ideal equipment, not always implemented with the best, most secure, most efficient design in mind. Whenever possible, I use free and open-source software (FOSS) tools for my customers, even “retrofitting around” pre-existing equipment when a redesign is not yet feasible.

With FOSS, community-driven development allows for unique and powerful features, equal to or greater than proprietary solutions. In a time where security should be of the highest priority, I like knowing that source code is free-and-clear to audit at any time by anyone in the world. FOSS can offer significant cost savings to clients, and many times, make doing a project that would otherwise be cost-prohibitive possible.

But misunderstood or misconfigured tech is often more problematic and insecure than not having it at all. I have a lab where I can test software, virtualization, networking and proprietary broadcast hardware and software, and I have remote access to broadcast equipment in the radio stations and labs of partners and clients all over the world.

Working remotely

The pandemic has taught us that remote work, even for broadcasters, is surprisingly doable. With so many at home — aside from audio transport, more on that later —



Writer
Chris
Fonte
Owner,
Fontastic LLC

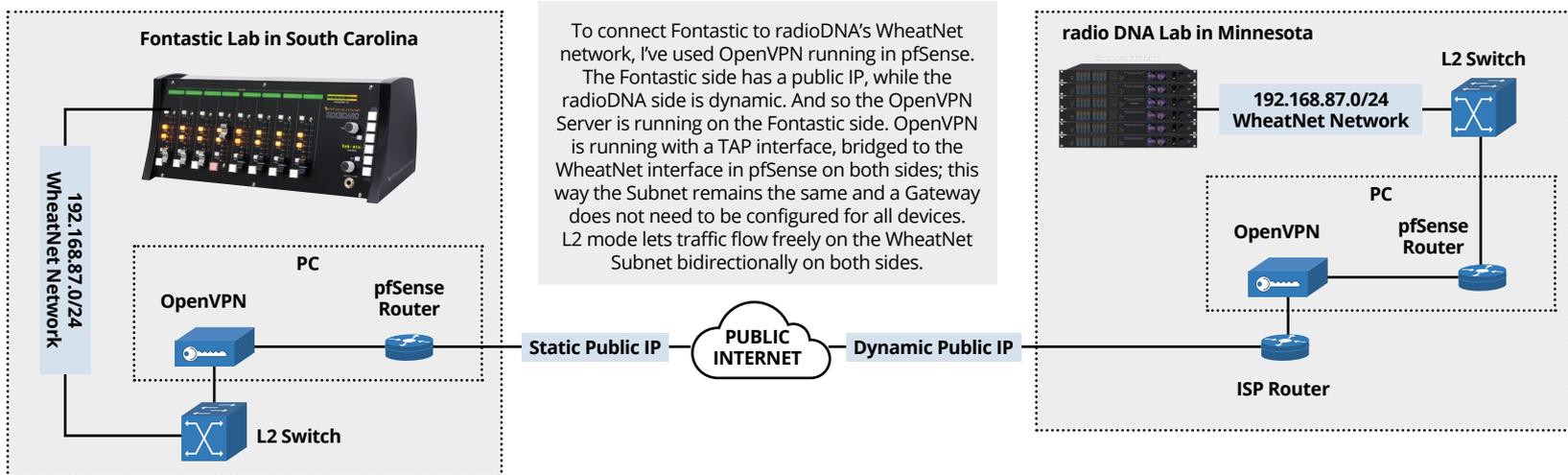
Above
WideOrbit
Automation for
Radio running on
a laptop at talent's
house connected
via OpenVPN, with
joint control of
the radio station,
audio from the
RodeCaster Pro
transported back
to the radio station
via Luci Live SE.

“IPSec,
OpenVPN
and
Wireguard are some of
the most common VPN
technologies, as well
as proprietary vendor-
specific ones.”

connecting to PCs station-side with tools like TeamViewer, VNC and LogMeIn have been the go-to solution.

But broadcast engineers like Jobie Sprinkle at WFAE in Charlotte, Henrik Poulsen at Radio NORDJYSKE in Aalborg, Denmark, and Tim Aquilina at 92.7 Mix FM in Maroochydore, Australia, have been looking for a way to give talent *physical* control again. And they are not alone.

Air talent around the world have grown up pushing physical buttons and adjusting physical faders and knobs



without needing to see what they are doing because of muscle-memory. Today, it is possible to build the creature comforts at home that talent has been accustomed to in studios for decades, while embracing new technologies never before possible. It can be done, even on a budget, by combining tools used by IT professionals across all industries with the plethora of tools already designed for broadcasters.

Security concerns

It has always been considered bad practice to open firewall ports to internal services across the public internet, especially when the data is unencrypted. But I still see it happening at broadcast facilities all over the world.

Today, security is of the highest priority, so, using encrypted VPNs with additional internal and external firewall rules in place are the way to go. IPSec, OpenVPN and Wireguard are some of the most common VPN technologies, as well as proprietary vendor-specific ones. SD-WAN, although a buzzword of late, is very powerful and allows for global networks of all kinds of devices.

Above
Fig. 1: Extending the network with a VPN tunnel.

In all cases, encryption at the highest level available should be used.

Getting into WheatNet remotely

Working from home myself, I wanted to test the reliability of remote physical control of gear from Wheatstone. In my home office lab sits a Wheatstone Sideboard. It is connected to a full Wheatstone AoIP system over 1,000 miles away. The Sideboard gives me real, physical, tactile control over a Utility Mixer — an 8-channel virtual console in a 1 RU Wheatstone Blade.

This is the intended purpose of a Sideboard, and it usually happens locally, but in this case, the Blade is in another time zone. With the Sideboard, I can select any source on the remote WheatNet system and make it available in the Utility Mixer I am accessing remotely.

So while I am controlling the mix locally, the mix itself is happening in the remote lab. Wheatstone supports using their automation control interface (ACI) between devices over a network, both locally and remote. The Sideboard is just one device on their list of control surfaces that can do this.

To make it possible in my lab, I have chosen a stack of FOSS tools including a router/firewall from pfSense with built-in OpenVPN. I set up an openVPN server on a static public IP address in my lab. On the remote-to-me lab's side of things, it is behind a consumer firewall with a dynamic IP address. Behind that firewall is a PC connected to the internal internet network on one Interface, and the WheatNet network on another. It runs an OpenVPN Client and connects over the internet to the server in my lab.

In OpenVPN on both sides, I am using TAP Interfaces, bridged to each local WheatNet network. On the tunnel, there is no routing happening, it's all layer 2, and so the devices on either side do not need a gateway defined to be able to talk to each other. In this way, I am effectively extending the same WheatNet Network across the VPN tunnel. See Fig. 1.

“With free and open-source software, community-driven development allows for unique and powerful features, equal to or greater than proprietary solutions.”

Climbing the firewall

Sometimes, due to time, budget or hardware constraints, setting up VPN tunnels between two sites via dedicated hardware is not always possible. This is where a newer technology called SD-WAN, or software-defined wide-area networking, can be helpful.

ZeroTier One, Nubula and Tailscale are examples of this and employ magic (UDP hole-punching) between firewalls to establish connections between devices on an internal network on either side, without the need to insecurely open firewall ports or use hardware VPNs. All data is encrypted end-to-end and allows bridging and routing securely through the internet behind dynamic IPs behind firewalls on both sides.

This is an exciting technology that is making waves across all industries, though setup and configuration can be a bit more daunting than standard VPNs.

Transporting the audio

Solutions for getting audio from point A to point B are in huge supply. Comrex, Tieline, Barix and others have complete lines of hardware devices that do this effectively and efficiently. And Wheatstone has the Blade 4 with audio codecs built in.

Real-time audio in the virtualized world — software to software, and software to hardware — has become the Holy Grail in our ongoing pandemic world, and products that do this have come a long way, too. Some are able to use the high-quality, freely available (and FOSS!) Opus codec via SIP-managed connections; others are web-based like ipDTL and Cleanfeed. LUCI Software offers solutions for mobile, PC, Mac and Linux (think LUCI Live and LUCI Studio). These can work via SIP or direct connection, and have become my go-to solution lately because of their immense flexibility and lifetime licensing fees.

Blurring the lines

Early this year, I heard from a client who wanted to hire a new afternoon talent, but she is located nearly 100 miles away from the radio station. Is this doable, and on a budget? The show would continue through the pandemic and beyond.

I got to work designing and implementing a cost-effective solution.

The station is a customer of WideOrbit automation for radio and runs version 4.0, part of the newest incarnation, and provides “joint control” of each radio station. To obtain this control remotely and securely, all that is needed is a VPN connection. I turned to my trusty pfSense/OpenVPN combo at the radio station, with an OpenVPN client running on a station-provided laptop at the talent’s house.

WideOrbit’s RadioClient, a native PC application, connects to the station-side RadioServers through the VPN tunnel, and the talent can control the radio station as if sitting in



More info

Fontastic LLC is a broadcast services company focusing on software and IT, helping radio stations around the world with integration projects.
Email: chris@fontasticllc.com
Twitter: fonte935

the studio. I configured workflows on hotkeys to route talent’s audio directly to air, take the feed off the air, send caller-audio to the talent, and route the backfeed to hear pre- and post-cut audio for voice tracking.

The off-the-shelf laptop is using a Rodecaster Pro for its audio interface, with an Electro-Voice RE20 plugged straight in. It has a solid DSP-based mic processor with a preset for the RE20, a listen and record bus and allows a mix-minus for the talent to hear return audio mixed with outgoing audio.

The client is particularly sensitive about subscription fees, so audio transport is handled via a one-time licensed version of LUCI Live SE on the laptop. This audio stream, along with WideOrbit control, is sent via the OpenVPN tunnel.

Station-side, a Windows virtual machine with an AoIP driver, is running LUCI Studio. LUCI allows different send and receive codecs, tailoring the codecs to the use-case.

In both directions, I wanted the lowest latency possible. I wanted return audio to be stereo, so that the talent feels like they are mixed well with the music. And so audio received from the talent uses a low-delay, mono, high-quality AAC codec. Return audio uses a stereo AAC codec with low delay and slightly lower quality. Currently, LUCI Studio is handling one bidirectional stream, but it is capable of 64.

Finally, phones are handled via Broadcast Bionics’ Caller One, running on a virtual machine at the station. Calls are answered via a web browser over the VPN remotely, with caller audio transported back to the talent via LUCI Studio, mixed on the Rodecaster Pro, sent back to the station as mix via LUCI Live SE, received by LUCI Studio, and recorded into WideOrbit as a produced package.

All of this does not *feel* remote for the talent, does not *sound* remote for the listener, and does not have an ongoing cost for the client.

Remote work is now a part of our lives across all industries. We live in a fantastic time of technology, where so much is available. And, now more than ever, it is possible for broadcasters, too. Doing it securely should be of the highest priority. 

“ All of this does not feel remote for the talent, does not sound remote for the listener, and does not have an ongoing cost for the client. ”