# VH2

Product Manual

## ABOUT COMREX

Comrex has been building reliable, high quality broadcast equipment since 1961. Our products are used daily in every part of the world by networks, stations and program producers.

Every product we manufacture has been carefully designed to function flawlessly, under the harshest conditions, over many years of use. Each unit we ship has been individually and thoroughly tested. Most items are available off the shelf, either directly from Comrex or from our stocking dealers.

Comrex stands behind its products. We promise that if you call us for technical assistance, you will talk directly with someone who knows about the equipment and will do everything possible to help you.

Our toll free number in North America is 800-237-1776. Product information along with engineering notes and user reports are available on our website at www.comrex.com. Our email address is info@comrex.com.

## WARRANTY AND DISCLAIMER

All equipment manufactured by Comrex Corporation is warranted by Comrex against defects in material and workmanship for one year from the date of original purchase, as verified by the return of the Warranty Registration Card. During the warranty period, we will repair or, at our option, replace at no charge a product that proves to be defective, provided you obtain return authorization from Comrex and return the product, shipping prepaid, to Comrex Corporation, 19 Pine Road, Devens, MA 01434 USA. For return authorization, contact Comrex by phone at 978-784-1776 or fax 978-784-1717.

This Warranty does not apply if the product has been damaged by accident or misuse or as the result of service or modification performed by anyone other than Comrex Corporation.

With the exception of the warranties set forth above, Comrex Corporation makes no other warranties, expressed or implied or statutory, including but not limited to warranties of merchantability and fitness for a particular purpose, which are hereby expressly disclaimed. In no event shall Comrex Corporation have any liability for indirect, consequential or punitive damages resulting from the use of this product.

## WHAT VH2 DOES

VH2 is a powerful hardware-based telephone system designed for broadcast on-air telephone management. Unlike legacy systems that interfaced to analog and digital dedicated phone lines, VH2 interfaces only to IP-based (VoIP or Voice-over-IP) telephone circuits. By this we mean that all phones lines are delivered virtually over a single Ethernet connection to the VH2.

The voice telephone industry has been moving toward IP voice service for many years already because it streamlines connections, enhances management, and reduces costs. IP-based voice carries these benefits for the broadcaster.

The VH2 (and VoIP in general) brings other advantages to broadcasters:

1. VoIP service is generally much less expensive than dedicated legacy phone lines.
2. VH2 implements a codec option (G.722) which delivers better fidelity (7KHz) than ordinary phone calls. This mode will be automatically selected when possible.
3. VH2 utilizes a VoIP protocol called SIP, which allows integration with existing and future VoIP gear (e.g. sharing lines and using extensions from PBXs).
4. VH2 can be easily interfaced to legacy PSTN, ISDN, T1 and E1 phone lines through the use of external gateway devices.

## ABOUT VH2 AUDIO PROCESSING

VH2's main function is to make your on-air phone calls sound great. From an audio perspective, it does this by:

- Making sure the caller you select is presented in the best possible audio quality to your console.
- Making sure that the caller can hear the studio host(s) clearly.
- Making sure if multiple callers are on-air, that they can hear each other clearly.
- Making sure callers with low or high levels are automatically equated (AGC).

A block diagram of VH2 audio connections is shown in **Figure 1** In this simple configuration, a feed of the station's program audio (pre-delay) is fed to the **HOLD AUDIO CH 1** port on the VH2. A studio "mix-minus" feed is attached to the **INPUT CH 1** port, and the caller(s) audio (for delivery to a single console fader) is available on the **OUTPUT CH 1** port.

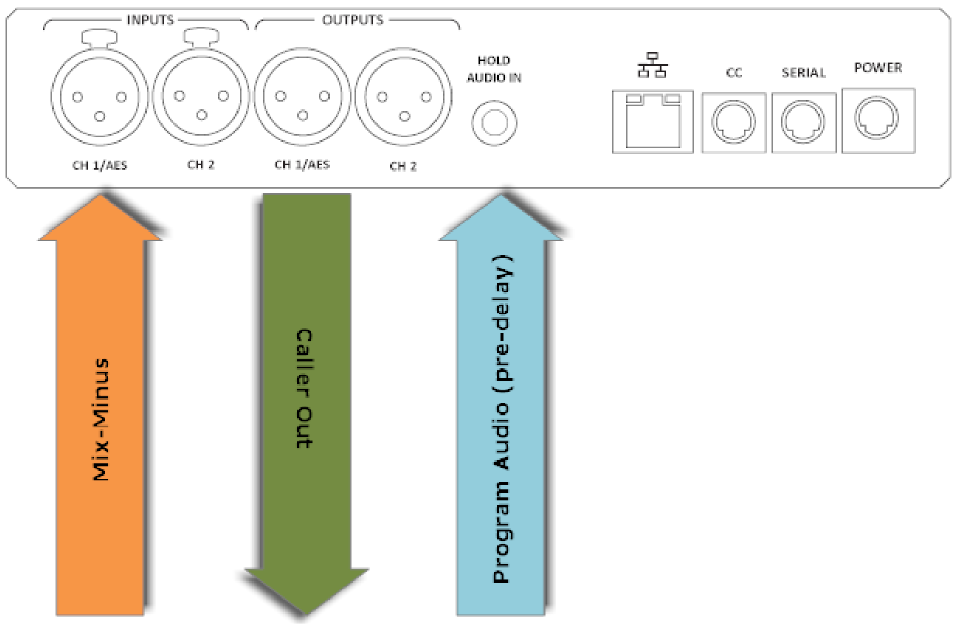



FIGURE 1 **VH2 SIMPLE AUDIO CONNECTIONS**

Many applications require that multiple callers be put on-air simultaneously. For this reason, VH2 contains two separate hybrids* that allow each caller to be presented on a different output from the VH2. This allows you to balance each caller on a separate console fader.

**Figure 2** shows this arrangement. Here, two caller outputs are used, but only a single mix-minus feed is applied to the **INPUT CH 1** port. This mix-minus must not contain any caller audio from either hybrid. An example of how to create this type of mix-minus is to use an auxiliary bus on your console, deselecting both telephone input ports from that bus.

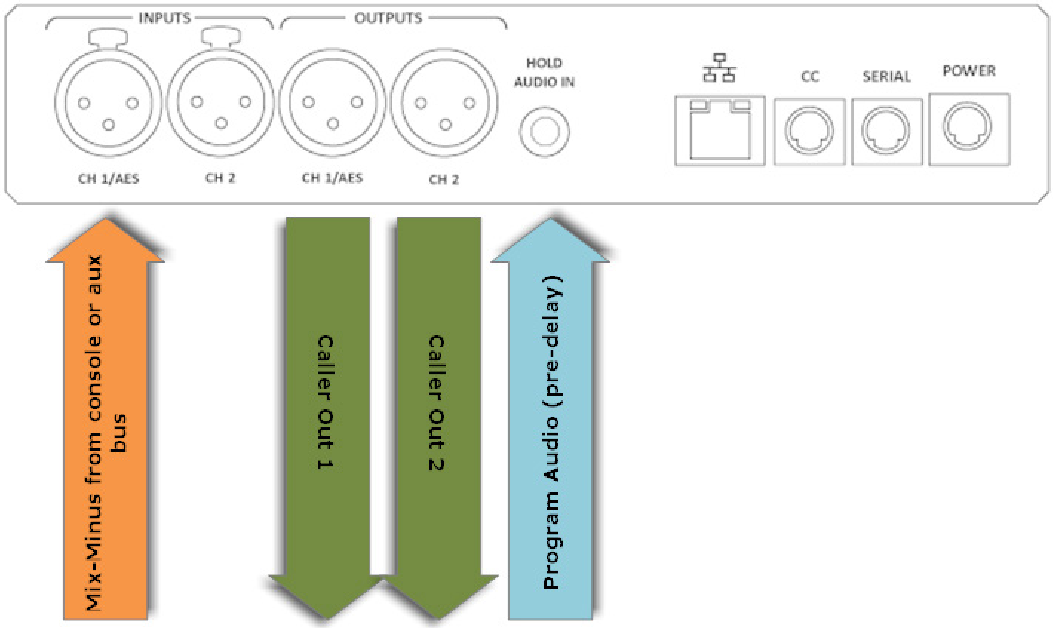



**FIGURE 2**

**The term Hybrid is a legacy from analog telephony products, which had the task of separating send and receive audio on the phone line. Since VoIP calls don't mix this audio, a Hybrid is technically not employed, but rather a conference is built for each output. For simplicity, we'll continue to use the legacy term for an on-air telephone channel.*

Some consoles have a dedicated "send" output for each fader associated with a caller. Often, the console will define these as "telephone busses" or "channels". In this case (**Figure 3**), each hybrid gets its own "send" audio from the console, each pre-configured with an appropriate mix-minus feed. These mix-minus feeds will be applied to the **INPUT CH 1** and (optionally) **INPUT CH 2** ports.

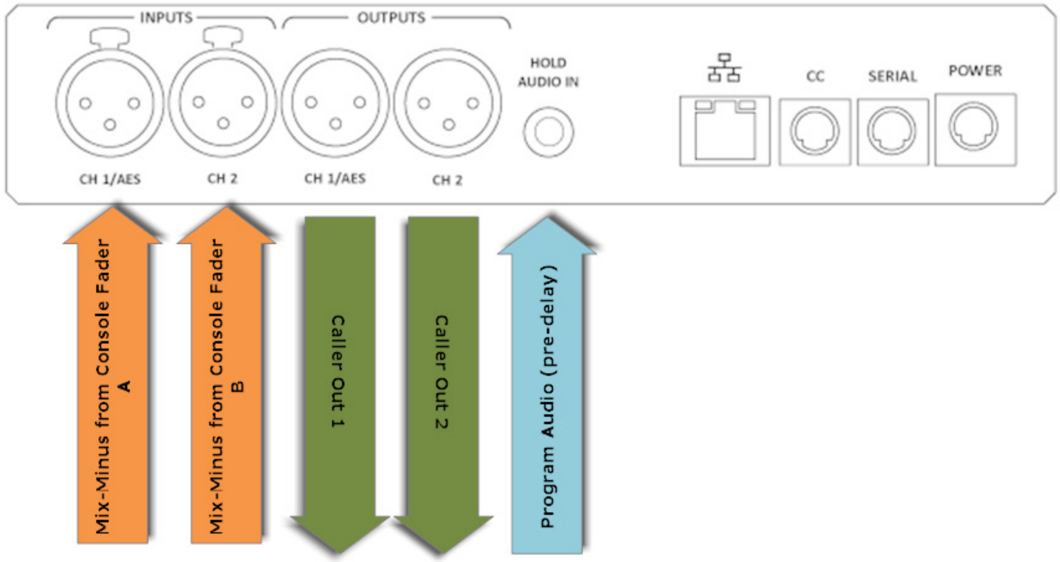



**FIGURE 3**

Finally, VH2 is capable of behaving like two, completely independent products for use in two, unrelated studios. This is shown in **Figure 4**.

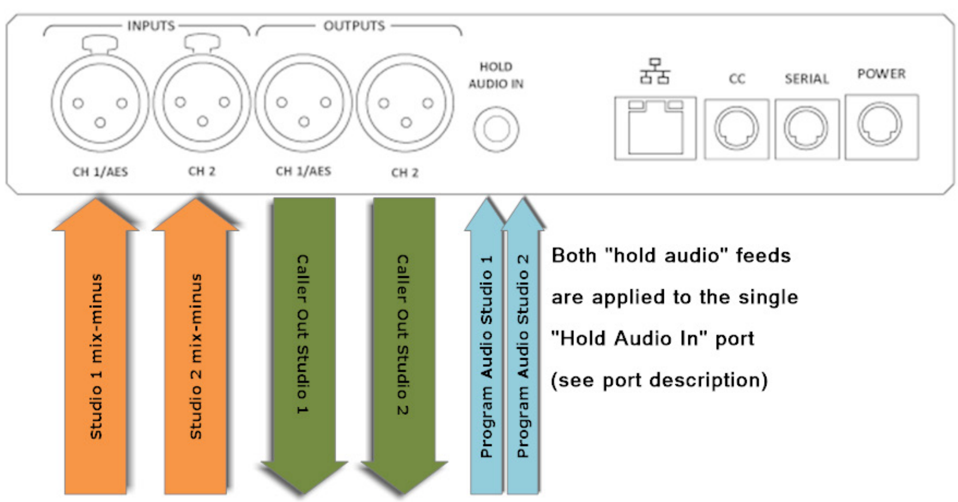



**FIGURE 4**

## ABOUT MIX-MINUS

Comrex support spends a lot of time discussing mix-minus, as it is not an obvious concept. In studio telephony integration, there is a golden rule:

***People connecting from outside the studio must never be sent their own audio back to themselves.***

If this rule is not followed, the result will be an unpleasant echo in the caller's ear. It is the responsibility of the studio tech to connect a feed to each return channel that is a mix of all important audio sources - in-studio mics, automation, carts, other remote sources, etc., minus the caller. Most modern consoles can do this easily. If not, other options exist and are discussed in the section "**Why Do I Hear Hear Myself Myself?**" (**Section XIV** in this manual).

Finally, when configuring and connecting for using both send inputs, the mix-minus feeds usually contains the audio from the opposite send channel. In instances where this is not the case, behavior can be configured to allow callers to hear each other within the VH2.

# II. SETTING UP VH2

## HARDWARE ATTACHMENTS

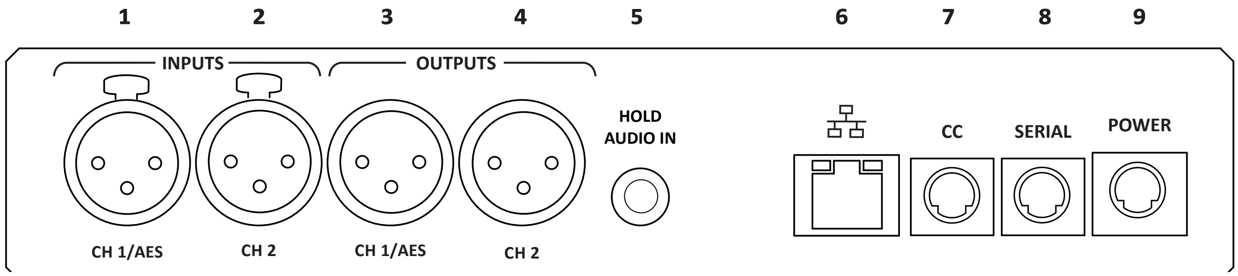



FIGURE 5 **VH2 REAR PANEL**

**Figure 5** shows the rear panel of VH2:

1 **CH 1 INPUT -** In analog mode, this XLR connector should be sent a balanced, 0dBu signal that is heard by callers when they are “on-air”. This is usually a “mix-minus” feed created on your studio console. In digital AES3 mode, both **CH 1 INPUT** and **CH 2 INPUT** are applied here (on left and right channel, respectively).

2 **CH 2 INPUT -** Use of **CH 2 INPUT** is optional. This analog input receives audio to be sent to callers on “hybrid 2” when “on-air”. This input is disabled when in AES3 mode or when configured for single “send” input.

3 **CH 1 OUTPUT -** This analog output will deliver the audio from callers who are sent to “channel 1” on air. Alternately, the system may be configured to present all caller audio on this single output. In digital AES3 mode, both **CH 1 OUTPUT** and **CH 2 OUTPUT** are delivered here (on left and right channel, respectively).

4 **CH 2 OUTPUT -** This analog output will deliver the audio from callers who are sent to “channel 2” on air. This output is disabled when in AES3 mode or when configured for a single “caller” output. Note special pinning of this 1/4” connector in the next section.

5 **HOLD AUDIO IN -** Analog only, route audio to this port that will be sent to callers “on-hold”.

6 **Ethernet Port -** Connect your network to this gigabit compatible Ethernet port.

7 **Contact Closure Connector -** This contains 4 contact closure input and output signals, for various remote control and tally functions as described further on.

8 **Serial Port -** This is an RS-232 connector that presents a serial port for future use.

9 **Power Supply input –** Apply the included AC desktop adapter output here. The power supply port requires 24VDC at 0.5A.

## AUDIO CONNECTIONS

All analog XLR audio inputs has a nominal level of 0dBu (full scale +20dBu).  When the input mode is changed (via DIP Switch #1) to AES3 mode, the AES3 output will rate lock to the input signal. Otherwise, AES3 output is 48KHz. AES3 inputs support 32, 44.1, and 48KHz sampling rates.

Analog input and output pinouts:

| | |
|---|---|
| Pin 1 | Ground |
| Pin 2 | Balanced audio + |
| Pin 3 | Balanced audio - |

AES3 input and output pinouts:

| | |
|---|---|
| Pin 1 | Ground |
| Pin 2 | Data + |
| Pin 3 | Data - |

The "**HOLD AUDIO IN**' port has a special pinout to be aware of. The 1/4" jack is designed to accept two signals, a "Hold" signal for channel 1 and channel 2. The inputs are unbalanced with a common ground.

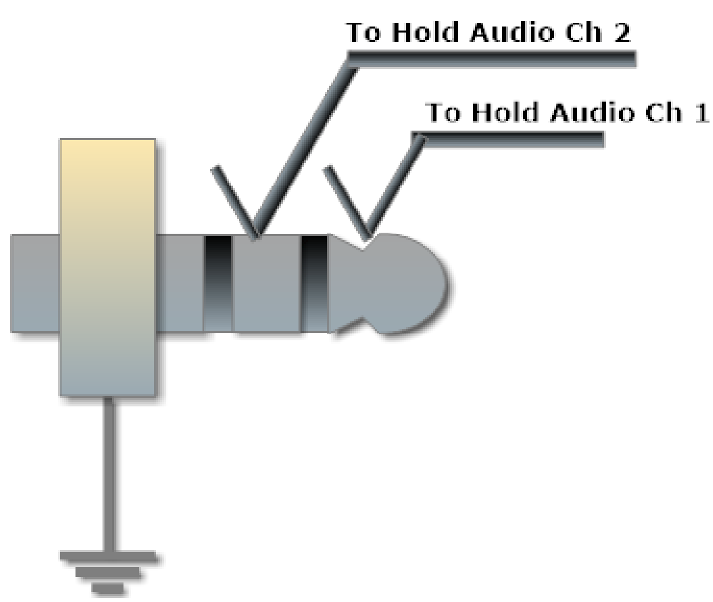
The pinout is:

Tip: Hold Audio in CH1

Ring: Hold Audio in CH2

Sleeve: Ground

Do not feed this port with actively balanced outputs, as it will ground one side of your active signal. If using only a single "Hold" audio source, connect the + side of the feed to Tip and Ring, and the Ground to Sleeve. If using two different "Hold" sources, connect CH 1 + to Tip, CH 2 + to Ring, and both grounds to Sleeve. Even though it's unbalanced, nominal level to this port is still 0dBu (+20dBu clip).



## CONTACT CLOSURES

Contact closure signals are available via the 9-pin mini-DIN connector.

The contact closure inputs parallel the action of the front panel buttons. The function of the output closures are set up in the behavior section of the configuration process, and can indicate different types of activity.

Inputs are triggered by shorting the respective input to Pin 9. Outputs consist of an open collector circuit which, when inactive, will offer a high-impedance path to Pin 9 and, when active, will offer a low-impedance path to Pin 9. These outputs are capable of sinking up to 200mA at a voltage up to 12V. Do not switch AC mains power using these contacts.

Contact Closure Pinouts:

| | | | | | |
|---|---|---|---|---|---|
| Pin 1 | Output #1 | Pin 4 | Output #4 | Pin 7 | Input #3 |
| Pin 2 | Output #2 | Pin 5 | Input #1 | Pin 8 | Input #4 |
| Pin 3 | Output #3 | Pin 6 | Input #2 | Pin 9 | Ground |

## MAKING CONNECTIONS

At a minimum, VH2 will need two audio connections and a network connection. Levels of all analog audio I/O is 0dBu (0.775V) nominal. This level will provide 20dB headroom before the clipping point. Input audio is reflected on the front panel LED based peak meters as indicated in **Figure 6**. Clipping is indicated by the red LED on these meters.

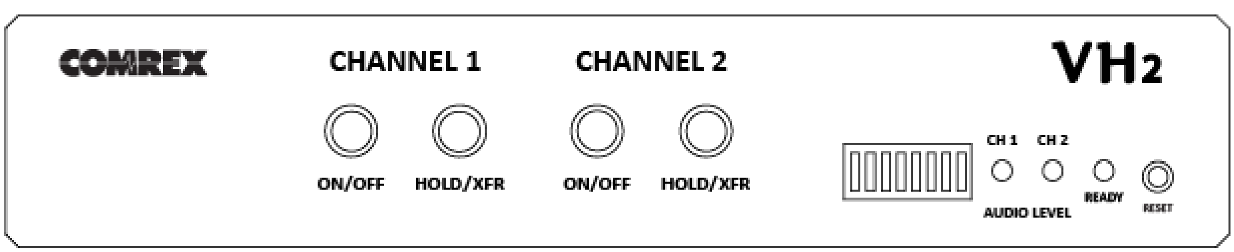



FIGURE 6 **VH2 FRONT PANEL**

VH2 needs a network connection to be useful. On VH2, network connections are made via a standard 1000Base-T Ethernet connection on an RJ-45 connector.

In most ways, VH2 will look like an ordinary computer to this network. In fact, VH2 contains an embedded computer with a Linux-based operating system and a full network protocol stack.

VH2 is perfectly capable of working over most LANs, but there may be situations where a LAN is heavily firewalled, subject to overloaded traffic conditions, or has security concerns. Better performance is possible if VH2 has its own Internet connection.

Since there may be bandwidth, firewall, and security concerns with installing VH2 on a managed LAN, it is recommended that your IT manager be consulted in these environments. The details that follow assume a working knowledge of IT topics and network configuration.

On VH2, the single Ethernet port handles connections to your phone lines and your web interface. IP addressing can be DHCP or Static, but if you are using the companion VoIP phone in your system, you will need to assign a Static IP address to the Ethernet port in order for the phone to find the VH2.

## ABOUT USING THE VH2 COMPANION PHONE

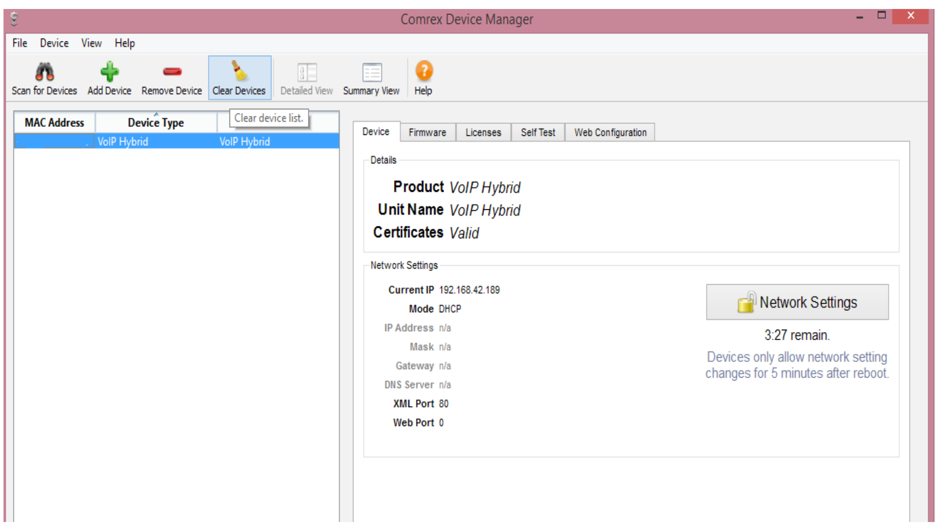
In a typical studio environment, it is convenient to be able to use a telephone set in conjunction with a telephone hybrid. This allows for outgoing calls to be placed, calls to be answered and screened, and for conversations to take place "off-air". VH2 supports this function with the approved VoIP companion phone (currently the **Polycom IP 331**). The VH2 can emulate the function of sharing a call between the hybrid and the phone, and "bouncing" a call between them with a single button push. The phone will require special settings (as outlined in section **IV. Configuring the VH2 Companion Phone**) to be configured as an extension of the VH2.

# III. CONFIGURING VH2 IP INFORMATION

VH2 is shipped from the factory set to DHCP mode, so it will find an address on your network if possible. The easiest way to find what that address is to use the Comrex **Device Manager** software, available on our website for both Windows and Mac platforms. You can also use **Device Manager** to upgrade the firmware on your VH2.

While it's not technically required for VH2 to have a static IP address, it is required for use with the companion phone, and if the DHCP address changes you'll need to find the new address in order to configure the VH2 via the Toolbox interface.

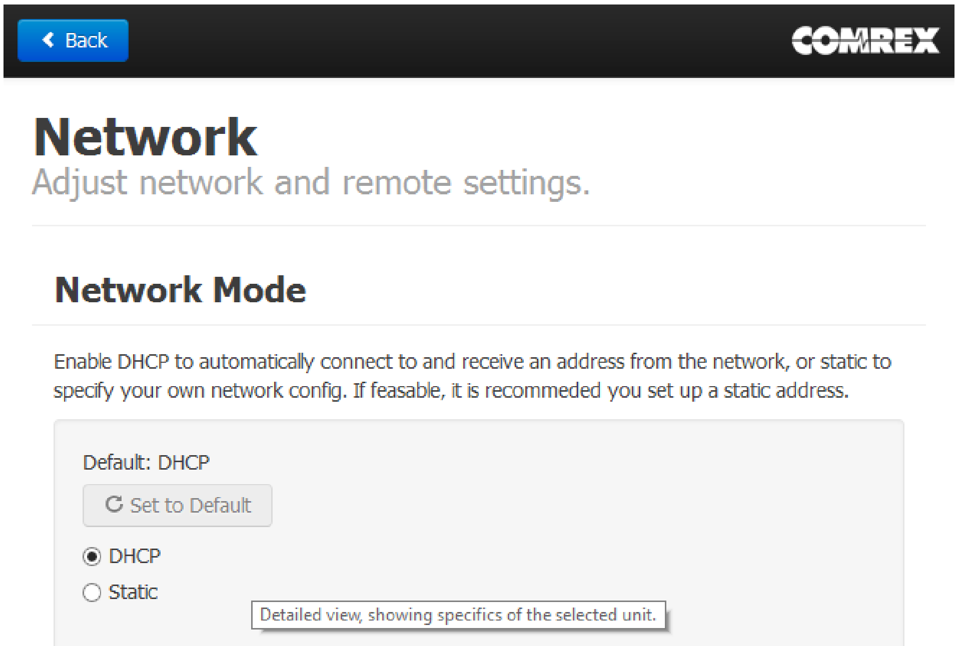
Install **Device Manager** on a computer on the same physical LAN as the VH2. Once installed and run, press the "**Scan for Devices**" button in the upper left corner to scan your network for Comrex products. The results should look like this:




There are two ways to set the static IP information using **Device Manager**. The "rescue" mode allows you to change parameters during the first five minutes of operation, and is initiated using the "**Network Settings**" button on the right pane. This mode is most useful if you've forgotten the unit password. The normal mode of changing any settings in VH2 is by clicking the "**Web Configuration**" tab in the right pane. This opens the unit's Toolbox interface. Toolbox is a web-based interface that allows you to set all relevant parameters of the product. Once you know the unit's IP address, Toolbox can also be accessed via a browser by going to **http://[ip-address]/cfg**.

Toolbox requires that you log on with a username and password. The default user name is “**admin**” and the default password is “**comrex**”. We recommend you change the default password, especially if your VH2 is exposed directly to the Internet.

Once logged in, choose the “**Network**” option. Under “**Network Mode**”, you’ll see the following options:




Change the mode to **Static**, and input the IP address, netmask, gateway and DNS information for your network. Click “**apply IP settings**” after all the information is correct. If you need to access Toolbox after this, you’ll need to rescan and log in again (or open a browser to the new IP address).

Alternately, to use the “Rescue Mode”, just click the **Network Settings** box on the right pane of **Device Manager**. If the 5 minute timer has not run out, a pop-up menu will appear allowing you to reconfigure the IP address, netmask, Gateway and DNS information and save it away. You will need to restart the VH2 for these changes to take effect.

# IV. CONFIGURING THE VH2 COMPANION PHONE

If you have the **Polycom 331 IP phone** for use with VH2, you'll need to set it up to be an extension of the VH2. The phone needs to be on the same physical LAN as the VH2, but it doesn't need a static IP address. You'll only need to program the static address of the VH2 into the phone.

Once the phone is on your LAN, it will get a DHCP address from the network. You can find what this is by pressing the menu button on the phone and navigating to the following using the direction cursor and enter key:

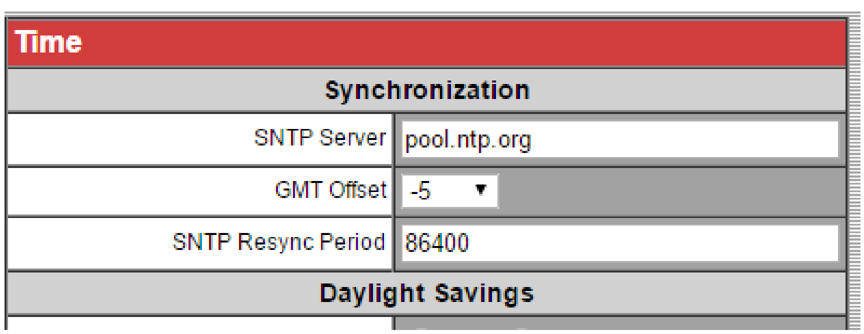
**2)Status-> 2)Network-> 1)TCP/IP Parameters**

Open a browser on your local computer and input the IP address of the phone into the URL bar. You will be prompted to log in to the phone's web interface. Default username is "**Polycom**" and default password is "**456**".

## TIME SETTING

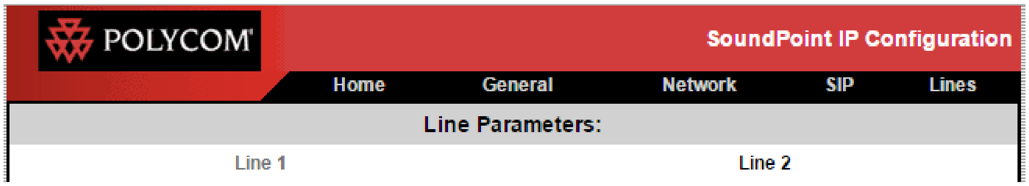
To stop the annoying time display from flashing an incorrect time and date, set a default SNTP server and time zone on the phone. Navigate to **General->Time** and set the fields to read the following:

**SNTP Server-> pool.ntp.org** (or any other SNTP server of your choosing)
**GMT Offset->** The offset of your time zone in hours from GMT (e.g. US Eastern = -5, US Pacific = -8)



## VH2 SETTINGS

To put the details of your VH2 into the phone, choose the **Lines** tab on the top menu of the configuration page:




Note the two sub-sections, **Line 1** and **Line 2**. You will need to program information into both of these pages.
On the Line 1 page, set the following fields:

Under *Identification*:

**Display Name -> 'Line 1'**

**Address-> '1100@<VH2_ip_address>'** (remove ' and <>)

**Authentication User ID -> '1100'**

**Authentication Password -> (default="456" - can be changed in Toolbox)**

**Label ->'Line 1'**

**Type->Private**

**Third Party Name->'none'**

**Number of Line Keys->1**

**Calls per Line->1**

Under *Server 1*:

**Address-><VH2_ip_address> (remove <>)**

**Port->5170**

**Transport->DNSnaptr**

**Expires->100**

**Register->1**

**Retry Timeout->0**

**Retry Maximum Count->3**

**Line Seize Timeout->30**

(leave the rest of the fields at default)

Your Line 1 Page should read like this (but with your VH2 IP address)

| Line 1 | |
|---|---|
| **Identification** | |
| Display Name | Line 1 |
| Address | 1100@192.168.42.15 |
| Authentication User ID | 1100 |
| Authentication Password | •••• |
| Label | Line 1 |
| Type | Private Shared |
| Third Party Name | none |
| Number Of Line Keys | 1 |
| Calls Per Line | 1 |
| **Server 1** | |
| Address | 192.168.42.15 |
| Port | 5170 |
| Transport | DNSnaptr |
| Expires | 100 |
| Register | 1 |
| Retry Timeout | 0 |
| Retry Maximum Count | 3 |
| Line Seize Timeout | 30 |

**Click Submit on the bottom of the page, then go to line 2**

On the Line 2 page, set the following fields:

Under *Identification*:

**Display Name -> ‘Line 2’**

**Address-> ‘1200@<VH2_ip_address>’ (remove ‘ and <>)**

**Authentication User ID -> ‘1200’**

**Authentication Password -> (default=”456” - can be changed in Toolbox)**

**Label ->’Line 2’**

**Type->Private**

**Third Party Name->’none’**

**Number of Line Keys->1**

**Calls per Line->1**

Under *Server 1*:

**Address-><VH2_ip_address> (remove <>)**

**Port->5170**

**Transport->DNSnaptr**

**Expires->100**

**Register->1**

**Retry Timeout->0**

**Retry Maximum Count->3**

**Line Seize Timeout->30**

(leave the rest of the fields at default)

Your Line 2 Page should read like this (but with your VH2 IP address)

| Line 2 | |
|---|---|
| **Identification** | |
| Display Name | Line 2 |
| Address | 1200@192.168.42.15 |
| Authentication User ID | 1200 |
| Authentication Password | •••• |
| Label | Line 2 |
| Type | Private Shared |
| Third Party Name | none |
| Number Of Line Keys | 1 |
| Calls Per Line | 1 |
| **Server 1** | |
| Address | 192.168.42.15 |
| Port | 5170 |
| Transport | DNSnaptr |
| Expires | 100 |
| Register | 1 |
| Retry Timeout | 0 |
| Retry Maximum Count | 3 |
| Line Seize Timeout | 30 |

**Click the Submit button on the bottom of the page.**

You are finished with the telephone setup.

# v. TELEPHONE CONNECTIONS

## INTRODUCTION TO SIP

SIP (Session Initialization Protocol) is the standard used by VH2 to talk to virtual phone lines. These lines must be created in some way externally before they are “applied” to VH2. “Applying” SIP lines to VH2 involves configuring the unit with certain information about the lines and the location of the server that delivers them. Then the unit can be allowed to register with the SIP server and automatically handle all of the interface with the virtual line.

The SIP line can be delivered from several sources:

1. A commercial telephony provider that delivers SIP-based lines over a public or private IP network.
2. A PBX that delivers telephony to extensions via SIP (in this case theVH2 is treated like a PBX extension).
3. A hardware gateway device designed to bridge legacy telephone services to SIP. These devices interface to the outside worlds via analog POTS/PSTN cables, T1/E1 circuits, or ISDN. They interface to VH2 via IP over Ethernet.

The setup and interface are identical for options 1 & 2 and similar for option 3.

The first step in setting up a SIP-based phone line is establishing an account with some kind of SIP-based provider or PBX. This process can’t be described here, but the result of this process is that you will have access to certain credentials for that account. These typically consist of:

**Server Domain** - The IP address or URL of the server to which you are being registered.

**Username** - The name that VH2 will use when logging into the service.

**Password** - The password associated with your account for security purposes.

In addition, several SIP providers require an **Authorization Username**, which is often defaulted to be the same as the username but sometimes is required to be different.

SIP is a complex protocol, and is based on Internet standards documents called RFCs. These documents are often subject to interpretation, and some PBXs, gateways, or telephony providers may have some obscure SIP settings requirements to work best with VH2. But most should work with just the credentials noted above.

Once VH2 is set up to work with a PBX or Internet Telephone Provider, it will automatically register with the correct server and maintain a connection to the server indefinitely. This way, the provider or PBX can notify VH2 of incoming calls to its phone number, as well as route outgoing calls correctly.

With hardware gateways, the process is reversed. VH2 will actually emulate a SIP server to the gateway device. The credentials will be entered into the gateway device. VH2 will only need to know the address of the gateway device

## SETTING UP A SIP PROVIDER OR PBX

SIP provider's info is entered using the web-based Toolbox Utility, accessible either by the **Comrex Device Manager** software (see **Network Settings**) or pointing a browser to **http://[ip_address]/cfg**. As a reminder, the default login credentials are **admin/comrex**.

In most arrangements, a SIP provider can deliver several telephone channels to you over a single account. When you subscribe to a SIP provider, make sure you understand how many simultaneous channels are provided with that account. These channels will act like a traditional telephone hunt group, so in the case of channel one being busy, calls will be routed to channel two, etc. You are only required to put in your credentials once for each account even if you have multiple channels.

Along with account credentials, you'll need a Direct Inward Dial (DID) number associated with your account. This is the "old fashioned" phone number users will dial to reach you. VH2 does not need to know this number - translation to the proper SIP channel happens behind the scenes at the SIP provider (although often the DID and SIP account name are the same).

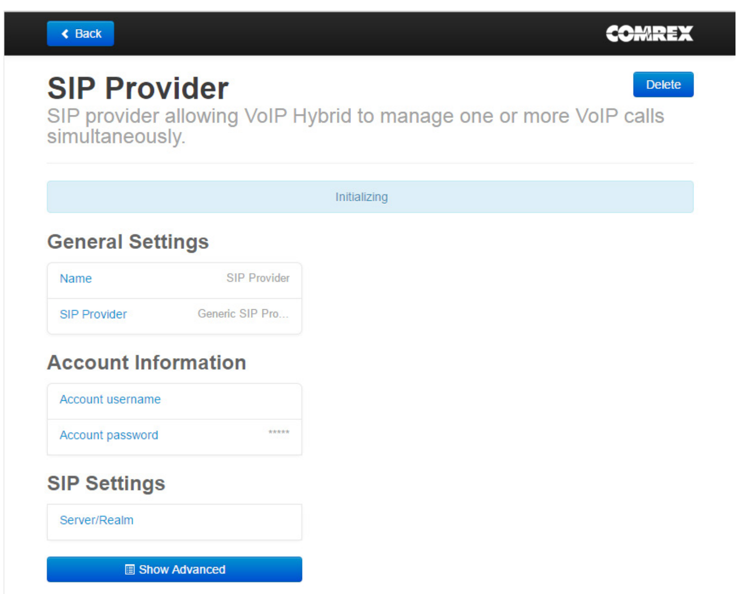



FIGURE 7

In Toolbox, navigate to **Channel Configuration->VoIP Providers->Add Provider-> SIP Provider**. The basic settings for a SIP provider are shown in **Figure 7**. First, you should give your provider a unique name in the "**Name**" field. This is used for reference only.

Under the SIP Provider entry, we've provided a list of commonly used providers. Using one of these profiles helps set up some of the more obscure settings that we've found necessary for these providers to work correctly. If your provider isn't on the list (or if you're setting up a SIP extension from a PBX) leave this set to "**Generic"**.

The next three fields are where you should enter your account credentials: **Account Username**, **Account Password**, and **Server/Realm**. These should have been provided to you by the SIP provider. **Account Username** is usually the name at the start of any SIP URI assigned. (More on that later). E.g., **username@sip_provider**. Make sure to enter only the username part.

The **Server/Realm** is the address of a registration server maintained by the provider. No web prefixes are needed here, just the address. E.g., **sip.comrex.com** or **iptel.org**. If you are programming a PBX extension here, this will be the IP address (or URL) of the PBX.

Under most circumstances this is all you should need. Setting these parameters, clicking **Back** then clicking **Restart** should start the process of VH2 registering with the provider or PBX. However, SIP registration can be tricky in some systems, and if registration fails you should check the required SIP settings carefully and use the **Advanced SIP** settings.

## ADVANCED SIP SETTINGS

By clicking **Show Advanced** in the **SIP Setup** menu, the list expands to include less used options:

### - UNDER ACCOUNT INFORMATION -

**Auth Username** - By default, this field is set internally to the same as the user.

**Name Entry** - Occasionally, SIP providers require this field be something different, and will outline that in their setup instructions.

## - CODEC SETTINGS -

Here is a list of the supported SIP codecs in VH2:

**G.711** - This is the same codec used in everyday normal telephone calls, and is the most common codec used in VoIP. When your SIP provider (or PBX) bridges incoming or outgoing calls to the legacy phone network, it will use this codec. So if you are using this mode primarily (e.g., not taking or making any wideband calls on this provider) this is often the best codec to use, so no transcoding needs to occur. The downside is that by modern standards, G.711 uses a lot of network data bandwidth (64kb/s). There are actually two flavors of G.711 (u-law and a-law) but VH2 automatically adapts between these.

**G.722** - This codec is the most common used in the “**HD Voice**” world to deliver wideband phone calls. It uses the same network bandwidth as G.711 (64kb/s), but delivers much higher fidelity (7KHz vs. 3KHz) sound.

The default behavior of the VH2 is the “normal” setting, where the unit will prefer wideband G.722 connections if possible, and drop to “telco quality” G.711 connections otherwise. Options are available to lock codec choice to G.722 or G.711 exclusively.

## - UNDER SIP SETTINGS -

**Outgoing Caller ID Name, Number** - If your provider or PBX allows changing the default name and number sent with outgoing calls, you can set these values in these two fields. Many providers will ignore this.

**Provider Binding Port** - This port is assigned by VH2 based on the number of providers you have assigned. Unless required, you should leave the default setting as is.

**Proxy Address** - Most providers use the same server address for incoming call registration and proxying SIP traffic. If your provider shows a different address for proxy, enter it here.

**Register Proxy Address** - Some providers require **REGISTER** messages be sent to a different server. If your provider has specifically required this, enter the address here.

**From Username** - SIP providers will usually automatically fill in the **From** field on an outgoing SIP call. If allowed, you can change the name used in this field by changing this entry.

**From Domain** - Likewise, the **From Domain** field is set automatically by the SIP provider for outgoing calls. If changeable, enter a desired name here.

**Expire Time, Retry Time** - These values determine how long to wait (without any communication) before the SIP provider will consider the registration connection lost, and once lost, how often to attempt to re-establish. The default values are usually best, unless strictly required to be changed by your provider.

**Register** - If you would like to save SIP provider entries for occasional use, you can set them to be disabled here by setting this value to “**No”**. No registration will be attempted until this setting is changed from “**No**” back to “**Yes**”.

**Register Transport** - Leave this set to “**UDP”** in all but very unusual circumstances.

**INVITE Contact Compatibility** - Optionally force extension to be sent in **Contact** field of **SIP INVITE**. This setting is required by **3Com NBX**.

**INVITE SDP Compatibility** - Optionally enable compatibility mode for **SIP INVITE lacking SDP**. This setting is required by **Cisco CUCM** and **3Com NBX**.

### - EXTRA SETTINGS -

**Outgoing Enabled** - Controls whether or not the line can make outgoing calls on this line using the companion VoIP phone.

## CONFIGURING FOR A PBX

In the case where you wish to set up incoming lines as extensions of an upstream PBX, the instructions are very similar. Your PBX will deliver channels to VH2 in the same way a SIP provider does, and you will need to set up the PBX and retrieve the proper credentials to program into the **SIP Provider** fields in VH2. IP PBX programming is usually very complex and is usually handled by the PBX vendor. Inform the vendor that you wish to set up an extension with “**x**” number of simultaneous channels and without additional features like conferencing, transfer, etc. (since VH2 doesn’t support these functions).

## GATEWAYS

Gateways allow you to use VH2 with traditional analog phone lines, as well as T1, BRI and PRI ISDN, and other legacy telephone trunks. Gateways will convert these telephone channels to SIP-style virtual phone lines. You will need to find gateways that deliver FXO style channels on their telco side - the ports on the gateway are designed to point toward the telephone service (and not interface with telephones and PBXs, like an FXS port does).

Setup of gateway devices can be quite complex. Comrex maintains some basic instructions on how to set the gateways we prefer on our website. **www.comrex.com**

Gateways deliver their virtual SIP lines differently than SIP providers. With gateways, VH2 becomes the "provider" and the gateways register with it. This means the credentials you set up on VH2 will be generated by you, and they will need to be mirrored into the gateway. To use a gateway, both the gateway and the VH2 must have Static IP addresses so they can find each other.

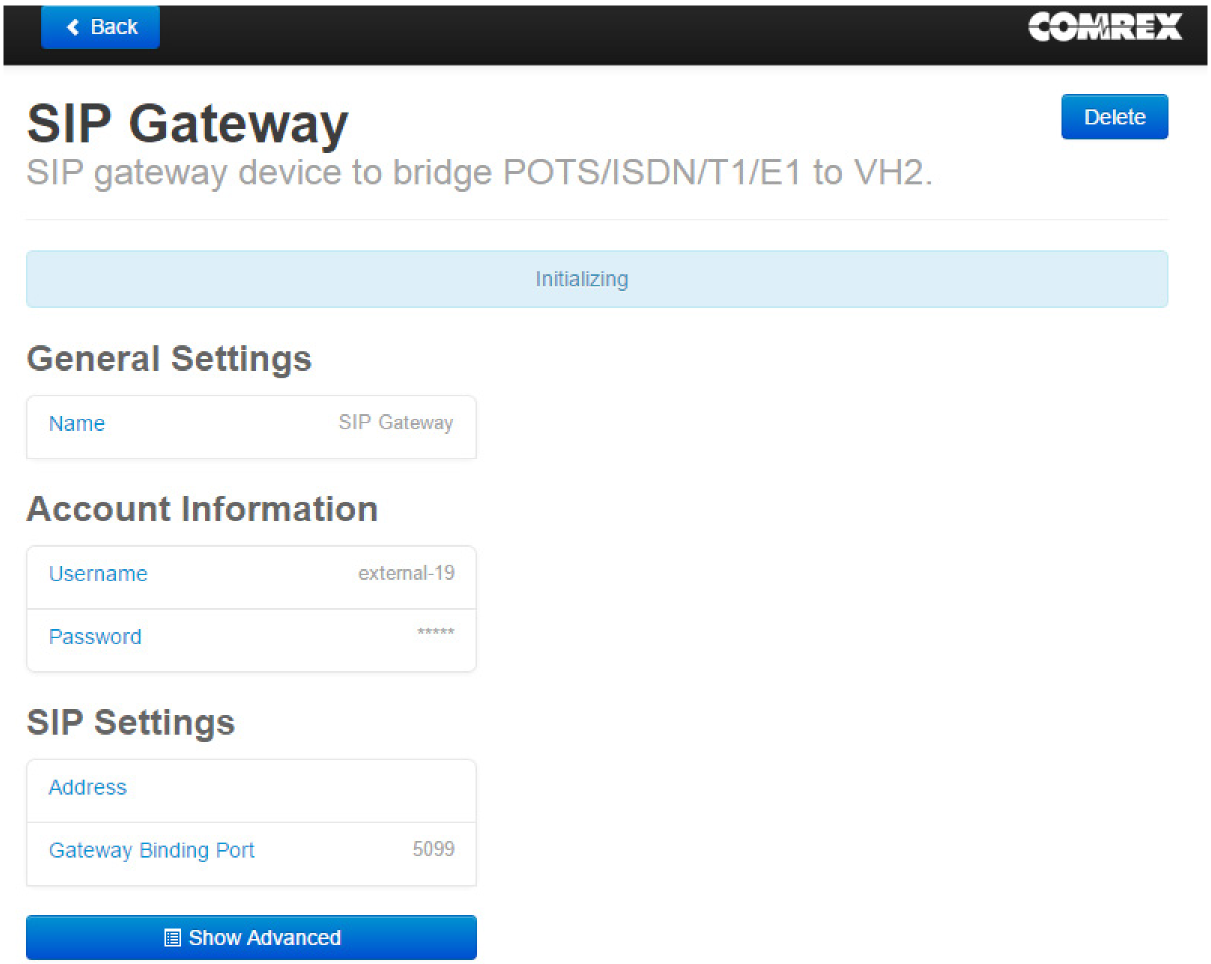



FIGURE 8 **NEW SIP GATEWAY**

**Figure 8** shows the settings for gateways (**Line Configuration > VoIP Providers > Add Providers > SIP Gateway Device**). Many of the settings are populated automatically by VH2, but can be changed to any value you wish.

**- GENERAL SETTINGS -**

**Name** - Give your gateway a unique name.

**- ACCOUNT INFORMATION -**

Username and Password - Locally generated values that the gateway will use to register to VH2

**- SIP SETTINGS -**

**Address** - The IP address of the gateway.

**Gateway Binding Port** - Automatically populated with an unused port. Must be mirrored into the **Gateway Settings**.

## ADVANCED GATEWAY SETTINGS

### DIAL PREFIX

Some Gateways require a certain prefix be dialed in order to select a particular legacy port for outgoing calls. If this is required for your Gateway (e.g. 991) enter it here.

### OUTGOING ENABLED

Allow outgoing calls on channels using this Gateway using the VoIP companion phone.

## SIP TRUNKS

SIP trunks differ from normal SIP providers in several ways. Rather than having VH2 “pull” the SIP channel from a provider, a SIP trunking provider will “push” the channel to a specific IP address of the user. This means in order to support SIP trunking you need a public, Static IP address, and no other devices can be utilizing the SIP ports at that address. Unlike normal SIP providers, only a single SIP trunk is supported on each VH2.

Most SIP trunk providers do not require registration information. Security is provided by the fact that they have initiated the SIP channel to the user’s address.

The setup of a SIP trunk is simpler than a normal provider. The option to add a new trunk is located in **Channel Configuration-> VoIP providers-> Add Provider -> SIP Trunk**. Once a new trunk is created, there will be no option to create another.

Although the settings menu for SIP trunking appears the same as for a normal provider, only a few of them are meaningful. You should put your trunking provider’s name into the **Name** field, choose your codec options (see normal provider settings) and set the correct SIP Port value (usually **5060**). If the trunk is active on your network, VH2 should sync with it automatically.

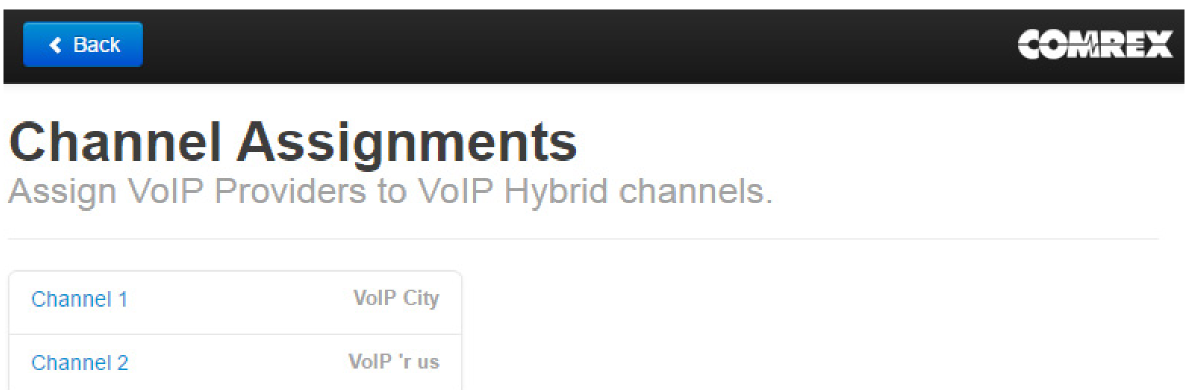
Very little feedback can be delivered on the status of SIP trunks by VH2. Trunks will show as “**registered**” as soon as they are created, regardless of actual status.

### LINE ASSIGNMENTS

Once you’re registered with at least one provider, gateway, or PBX, you can assign the VH2’s “channels” to that provider.

**Important: Simply registering is not enough. VH2’s channel(s) must be assigned to your provider(s) before you can use it.**

You can have up to two providers, and each provider can be assigned to one or both channels. Channel assignment is done in Toolbox via **Channel Configuration->Channel Assignments**.




The figure shows each channel assigned to a different provider. Each channel could also be assigned to the same provider, and the usual behavior is for calls to “hunt” between the channels (first call on channel 1, second on 2).

# VI. SYSTEM BEHAVIOR

Back at the main Toolbox page, selecting **System Behavior** allows you to set how the VH2's audio I/O interacts with your studio. The options are:

## AUTO-ANSWER

You can individually set each channel to auto-answer or not. If set to auto-answer, you can choose whether each channel puts each call to "on-air" state or "on-hold" state.

## STUDIO AUDIO I/O

Here you can select how many input and output audio ports are used on the VH2. The choices are:

a. **Single In/Single Out** - Only the **CH 1 INPUT** and **CH 1 OUTPUT** are used. All callers hear **CH 1 INPUT** and all callers audio is output on **CH 1 OUTPUT**.
b. **Single In/Dual Out** - The **CH 1 INPUT** is used, but both **CH 1 OUTPUT** and **CH 2 OUTPUT** are used. Caller audio appears separately on these channels (to be sent to two separate console faders, for example). Both callers hear audio sent to the **CH 1 INPUT**.
c. **Dual In/Dual Out** - Both **CH 1 INPUT** and **CH 2 INPUT** as well as **CH 1 OUTPUT** and **CH 2 OUTPUT** are used. Caller 1 hears **CH 1 INPUT** and appears on **CH 1 OUTPUT**. Caller 2 hears **CH 2 INPUT** and appears on **CH 2 OUTPUT**.

Note that none of these settings has any effect on whether callers hear each other. That function is chosen in the next option.

## ABOUT MIX-MINUS

When we refer to "send" audio to the caller, we're talking about the feed that is attached to the VH2 **CH 1 INPUT and CH 2 INPUT**. This is the audio that the caller hears when "on-air". It is essential that these feeds are specially mixed so that the caller output is not part of that mix. Many consoles provide for a special audio bus especially for this function. On some other consoles, you can use the secondary or audition bus (with callers deselected) to create this feed. But under no circumstances must incoming callers hear themselves sent back, or they will experience an annoying echo.

## CALLER MIX

Determines whether the callers can hear each other in the "conference" built within the VH2. If your console provides individual mix-minus for telephone channels, this should be off, as the mix will be done in your console. If the console doesn't handle this, and you are using a single mix-minus for both channels, turn this function on so the callers will be conferenced within the VH2.

## AGC

Selects whether VH2 applies AGC to the caller outputs, helping minimize large level changes between calls.

## CALLER DUCKING

Selects whether VH2 applies an algorithm to the caller audio to reduce it when the host voice is detected. This allows the host to "dominate" the conversation. Ducking, if enabled, has three choices (**Low**, **Medium**, **High**) that allows selection of how much the caller is reduced when the host speaks.

## CALLER ON-AIR TONE

Selects whether the caller hears a short "beep" when put "on-air". This is sent to the caller only and is not heard "on-air".

## CONTACT CLOSURES

The VH2 has four input and four output contact closures available.

**Outputs** -This entry determines the function of the four outputs. The choices are:

1. **No Function**
2. **Call On Air** - Closed when a call has been put "on-air" on any line.
3. **Call On Air 1 & 2** - A call has been placed "on-air" and is active on either the **CH 1 OUTPUT** or **CH 2 OUTPUT** specifically.
4. **Call Ringing** - An incoming call is ringing on any configured line.

**Inputs - Momentary/Latching**

The four contact closure inputs parallel the four front panel **On/Off** and **Hold Xfr** buttons. Since these buttons act as "momentary" toggles, this is the default action of the contact closures as well (e.g. a momentary closure on cc input 1 engages the On/Off CH 1 function, and a secondary closure disengages it). Some remote control systems can only deliver "latched" signals (closed for on, open for off). Changing this setting to "**Latching**" allows the VH2 to be remote controlled in this way. It does not have any effect on how the front panel buttons operate.

Note that the "**Xfr**" function on the "**Hold/Xfr**" button requires a momentary "long press" of the button to operate. Because this can't be supported in "latched" mode, the "**Xfr**" function is not available by remote control in latched mode.

## TEST MODES

These options are used for diagnostics or demo purposes only and will interfere with normal operation when enabled. Make sure all test modes are off before operating VH2 normally.

- **Audio Test** - These options provide for specific audio paths to be enabled (e.g., **CH 1 INPUT** to **CH 1 OUTPUT**). These are used in unit production tests and can also be used to troubleshoot general hookup issues. Modes are also offered that generate a tone from the caller out ports.
- **Contact Closure Test** - Likewise, enabling this option puts the contact closure feature in "loopback" mode with inputs directly driving outputs (e.g., **CH 1 INPUT** to **CH 1 OUTPUT**).
- **LED Test Mode** - tests the function of the front panel button and LED indicators

# VII. NETWORK CONFIGURATION PAGE

The **Network Configuration** page is shown in **Figure 9**. The settings in this section determine the IP settings of the Ethernet port, along with which network services are active on the VH2.

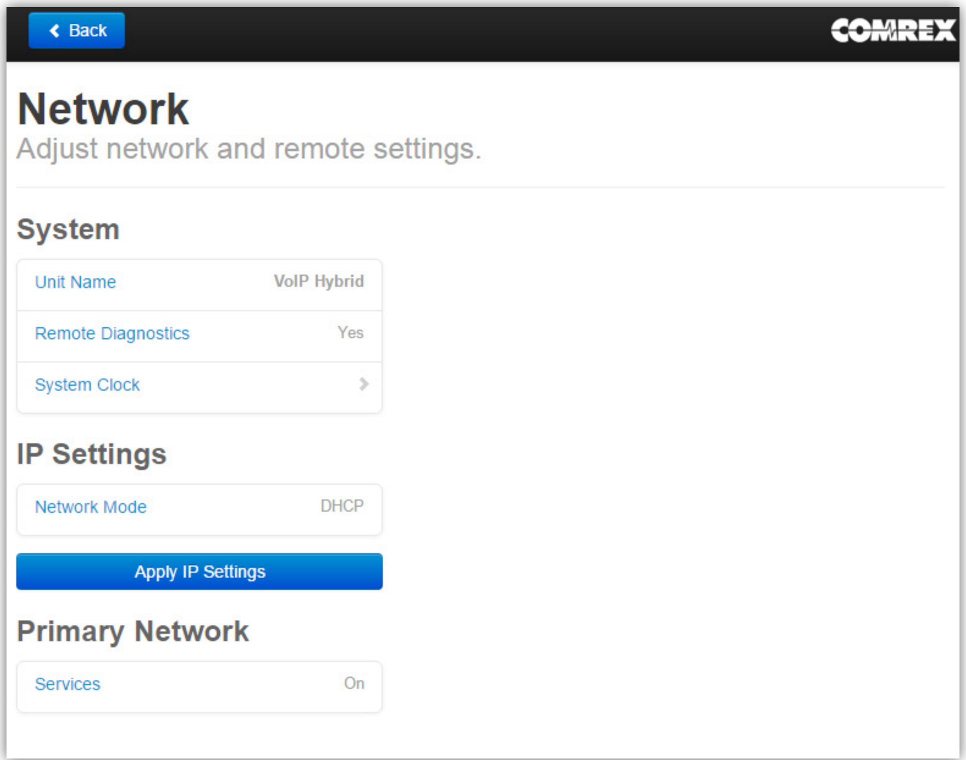



FIGURE 9

## - SYSTEM -

**Unit Name** - Giving your VH2 a unique name will help you identify it on the network.

**Remote Diagnostics** - Comrex support has the ability to connect to your VH2 via the SSH protocol to troubleshoot issues. This requires a private keypair that we don't provide. If you have security concerns about SSH, you can disable it by setting this option to "**No**".

**System Clock** - VH2 maintains a network connection to an NTP server that delivers the time-of-day information for use by system logs. The specifics of that function can be changed here. The default settings allow for a pool of public servers to be used.

## - IP SETTINGS -

As described in the previous section "**Configuring VH2 IP Information**", this is where you set the static IP address of the VH2's main Ethernet port.

## - PRIMARY NETWORK -

**Services** - Using this option, you can secure VH2 so that the Toolbox interface is no longer delivered via the Ethernet port. Note that once this is turned off, no further configuration of VH2 is possible. To change any setting you must first apply a factory reset to the VH2, which will wipe all its settings, including all your VoIP account info and static IPs. Turning services off also disables your VH2's ability to sync with the Comrex Device Manager (until a factory reset is performed).

# VIII. SECURITY AND USER ACCOUNTS

Security settings are available in the Toolbox web-based configuration page under **Security**.

The following options will be available under **Security**:

### HANDSET SIP PASSWORD

The VoIP companion phone available for the VH2 requires a password login in order for VH2 to communicate with it. The default password is “**456”**. This is also the password used to gain access to the full configuration of the phone via the web or the keys on the phone. If you wish to change this password from the default, you can let VH2 know the new value in this field. Comrex recommends changing this password.

### CHANGING THE PASSWORD ON THE ADMIN ACCOUNT

It is highly recommended to change the password on VH2 to prevent undesired configuration changes. This is done under **Security->Accounts->Admin** in Toolbox. Note that once the password is changed, it must be known to access Toolbox from a web browser or the **Device Manager** software. If the password is forgotten, a full factory reset will be required, clearing all settings including static IP and telephone provider settings.

# IX. FACTORY RESET

## FACTORY RESET

VH2 can be restored to factory settings, clearing all IP, behavior and provider info, two ways:

1. If the password is known, use the **Device Manager** software to login to VH2. Select **Device->Reset to Factory Defaults** to issue the reset command.
2. If the password is not known, you must issue a hardware factory reset. This is done via the following sequence:
    1. Put dip switch 7 & 8 up.
    2. Press the reset button once, wait until the **CH 1 CH 2** and **Power** LED flash red and green.
    3. Put dip switch 7 & 8 down.
    4. Press the reset button once.

## DIP SWITCH SETTINGS

The front panel DIP switches have the following function:

1 Analog in (down) / AES3 in (up)

2 Analog out (down) / AES3 out (up)

3 Unused

4 Level Indicators reflect Send audio (down)

Level Indicators reflect Caller audio (up)

5, 6 Unused

7, 8 Factory Reset (see previous section)

## ON/OFF BUTTON (CH 1 AND 2)

This is a toggle which determines if the call is active on the selected channel. When pressed, an incoming call will be answered. This button will also “grab” a currently active call on the companion telephone and place it on the desired channel. Calls selected this way are considered “on-air” and caller audio is routed via the selected behavior settings. Pressing this button when a call is active drops the call, whether “on-air” or “on-hold”.

## HOLD/XFR (CH 1 AND 2)

This is a toggle that places an active call “on-hold” from the “on-air” state. The caller will be removed from the main audio ports, and hear only the audio presented to the VH2 “on-hold” input. Pressing this button while an incoming call is ringing will send the caller directly to “on-hold” state. Pressing this button while a call is “on-hold” will send the call back to “on-air” state.

If a call is active (either “on-hold” or “on-air”) a long press of this button (over two seconds) will transfer the call to the companion telephone set. The set will ring, but the call will remain in its previous state (“on-hold” or “on-air”) until the handset is answered.

## INDICATIONS

The ON/OFF and HOLD/XFR buttons are lit to indicate the state of each particular channel

| ON/OFF | HOLD/XFR | Function |
|---|---|---|
| Off | Off | Idle-registered and ready for call |
| Blink Green | Off | Ringing |
| Green | Off | On-Air |
| Green | Green | Hold |
| Blink Green | Blink Green | On Handset |
| Red | Off | Auto-Answer |
| Red | Red | No Provider Assigned |
| Blink Red | Off | Provider Assigned, but not registered |

The “Ready” Indicator is used to signal the overall network condition. Its states are:

**Off** - System is starting up

**Fast Blink** - Loading application

**Red** - No Internet connectivity detected

**Green** - System has IP address on network

## GENERAL OPERATION

### INCOMING CALLS

Calls will ring the handset (if used) and blink the **ON/OFF** indicator for the channel with the incoming call. Calls can be answered via the handset, or placed directly “on-air” or “on-hold” via the front panel button. Calls on the handset can be put “on-air” or “on-hold” via the same buttons.

### OUTGOING CALLS

Place calls on either channel 1 or 2 via the handset. Place them “on-air” or “on-hold” with the appropriate channel buttons.

### TRANSFERRING TO HANDSET

From “on-air” or “on-hold”, long press the **HOLD/XFR** button. The handset will ring. Answer it and the call will be transferred (note the call is still in the current state until handset is answered).

### ENDING CALLS

From “on-air” or “on-hold” state, press the **ON/OFF** button to end the call. From the handset, simply hang up.

# XI. ABOUT SIP

**NOTE: This section is taken directly from our VoIP Primer. You can view and download the primer at www.comrex.com.**

## Introduction

At Comrex, it's our job to keep ahead of new and intriguing technologies that we can leverage for our customer, the broadcaster. But it's important that as we ride the wave of new tech, we don't forget about the people in our industry who have "stuff to get done", and can't afford to spend hours reading about all the newest developments.

We've found this to be the case in recent years with the introduction of ISDN, POTS codecs, and IP audio codecs. In each case, we decided to put together a "primer" for those who wished to gain the knowledge needed to use these tools effectively, but were short on time. The goal was to put together all the vital information in a booklet that could be consumed in under an hour. The feedback we've received has proven these efforts to be worthwhile.

A new disruptive technology is taking hold, and it's now time to cut another primer. Due to cost and necessity, broadcasters are finding they need to learn about Voice over IP (VoIP), and they need to learn fast.

**This booklet provides some basics about VoIP in an easily digestible form.**

VoIP provides a way for computer networks and other devices to emulate traditional phones and phone lines. Most modern business PBX systems have migrated to VoIP already. In some circumstances, legacy phone lines (PSTN or POTS) are no longer available and VoIP is the only choice.

Like a traditional line, a VoIP link consists of a service provider and an end user who owns a telephone instrument. But in this case, the provider is based in the "cloud". Alternately, the VoIP lines can be delivered from an upstream PBX. The end-user gear is a specialized VoIP telephone, or software running on a PC or mobile device that performs the same functions.

The Comrex STAC VIP is a sample of a device designed to interface with VoIP service. It can handle six or twelve calls simultaneously and provide the typical screening, audio processing, and control functions expected of broadcast call-in systems. For users with less call volume, the VH2 Hybrid is a dual-channel VoIP-to-studio interface. In addition, all Comrex IP codecs like ACCESS and BRIC-Link can communicate over standard VoIP protocols.

## IP Concepts you need to know

If you're already an expert on IP networking concepts in general, feel free to skip to the next section about RTP. But here are a few basic concepts you'll need to master to continue learning. This is much less than a complete overview of IP networking--only concepts directly relevant to VoIP are covered.

### IP basics

IP is short for Internet Protocol, but it doesn't always pertain to the Internet (as in, the public version). In a nutshell, IP networking involves creating packets of data, attaching certain headers to specify contents and assign addresses, and applying them in sequence to some kind of network capable of transmitting them. Physically, the network is usually Ethernet, although it may be Wi-Fi, 3G, satellite, or lots of other mediums.

## Addressing

Devices connected to an IP network are dealt an "IP Address". Under the IPv4 protocol (the most widely implemented), this address consists of a 32-bit numeric value. Putting on your "binary thinking cap", this can also be thought of as four 8-bit bytes. A byte can have a value from 0-255, so IP addresses are usually written as a sequence of four decimal numbers (separated by dots) like 192.168.0.23 with each integer having an upper limit of 255.

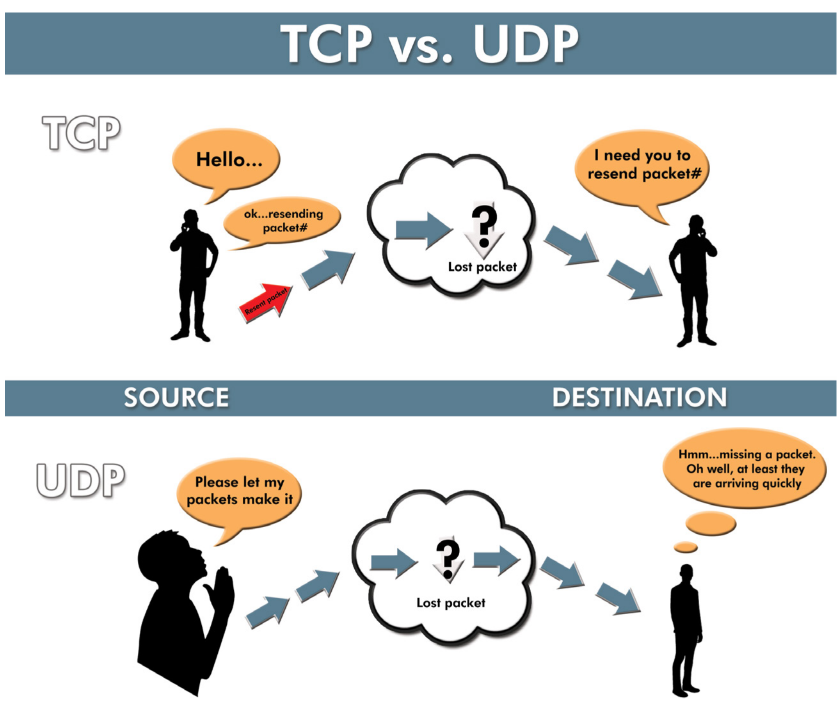
## Ports

The IP address is the main identifier used to specify a destination to send packets to within a network. But since IP compatible devices can make simultaneous connections for different reasons (e.g. web surfing and email), a scheme is used to designate a specific "port" on a machine, which is essentially a 16-bit sub-address contained within the header of the packet. These ports are usually written as simple decimal values (e.g. 80, 5060), and traffic sent to a specific port on a machine can only be accessed by a program or service "listening" on that port.

## TCP vs. UDP

The most common types of IP traffic fall in two sub-categories, TCP/IP and UDP/IP. The difference is important. Most web-related traffic travels via TCP, which has built in mechanisms for integrity checking and error-correction. This means that if the TCP "stack" within a machine has delivered a packet from the network, the packet is guaranteed to be correct, and if lost will be resent.It might surprise you to know that it's not TCP that's used for most real-time media on the web. This is because TCP has quite a bit of overhead in terms of data, and can easily add time delays if packets get corrupted.

VoIP and other real-time communication protocols use UDP, which is a much simpler delivery method. There is no error correction or resending available at the native UDP layer. UDP is sometimes referred to as the "send and pray" method, since the network provides no guarantees of delivery of any kind. In it's simplicity, UDP is a better choice for real-time communications because higher-level applications can be designed to make smart choices about error protection vs. delay.



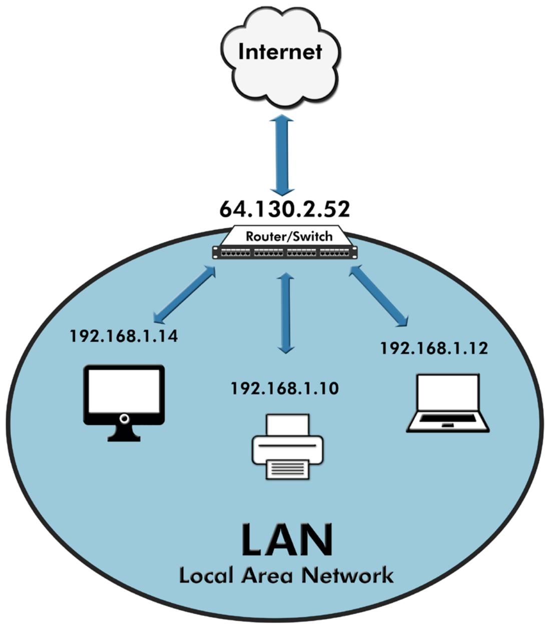
Packets sent on IP networks will include a destination IP address/port combination, and a source IP address/port combination. These act like the destination and return address on an envelope, and allow the packets to be responded to over the network.

The destination port is the most important to IT people, as it's the one that they need to be sure is open to receiving communications. When IT folks refer to a service as "running on port x" they are referring to the destination port.

We designate an IP connection via its protocol, destination IP address, and port combination in this form: **<protocol> <destination address:port>** e.g. **UDP 192.168.0.7:5060**

## LAN vs. Internet

Most of the networking you'll be dealing with will exist within your LAN (Local Area Network) and connections between devices within the LAN follow ordinary rules to send packets between each other. But in the situation where you wish to connect to a device outside the LAN (which is most common) special rules need to be followed.




LANs have IP addressing conventions that allow a range of addresses to be reused within the network, and prohibit those addresses not to be used again on the public internet. This allows for many devices to site behind a router, which has a single internet (publicly addressable) IP address, and each LAN device to have a private, reusable IP address. By convention the address ranges start with the digits 192.168.x.x, 172.16.x.x, or 10.0.x.x. So, for example, if a machine tries to connect to another at an address of 10.0.0.75, it is necessarily trying to send packets only within its LAN. The range of addressable LAN addresses is called a subnet, and must be programmed into each machine using a subnet mask entry.

If a machine on a LAN wishes to send packets outside the subnet, it must communicate with a gateway (usually a router) at a fixed IP address.

### Network Address Translation

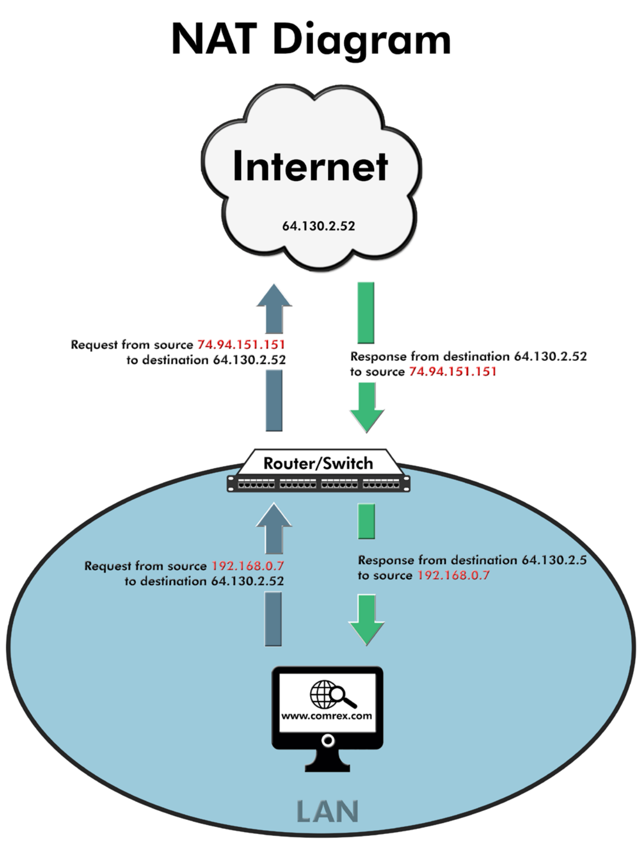
The concept of how a gateway router provides translation services to the Internet is extremely important in the field of VoIP, if only because it causes so many headaches. Known as Network Address Translation (NAT), it's easiest to use a diagram to illustrate a typical gateway scenario describing a user on a LAN accessing a web page at comrex.com. For this illustration, we'll ignore the concepts of DNS and URLs (which aren't particularly useful for VoIP) and live the fantasy that the user is accessing the comrex.com page via its public IP address, which is (as of this writing) 64.130.2.52. In our scenario, the user has a laptop on a LAN using the popular 192.168.0.x subnet addressing scheme, and specifically has the address of 192.168.0.42 assigned to it.

The user will input the web page address into his browser, and the computer will recognize the address as outside the subnet it has been programmed to work on. So it will form a packet, whose payload consists of a request to view the web page, and hand it to the gateway router, which is located at the local address programmed into the laptop (192.168.0.1).

Because the router is acting as a gateway, it actually has two IP addresses. The LAN address (192.168.0.1) is used by devices on the LAN. The WAN address (74.94.151.151) is the address assigned by the Internet Service provider. This address is public, in that it is addressable by every device on earth that is connected to the Internet.

The router will record the source address of the packet (192.168.0.7), change it to the public IP of the router (74.94.151.151), and send it along to the destination IP address. This is so the web site knows the correct address to which to respond.

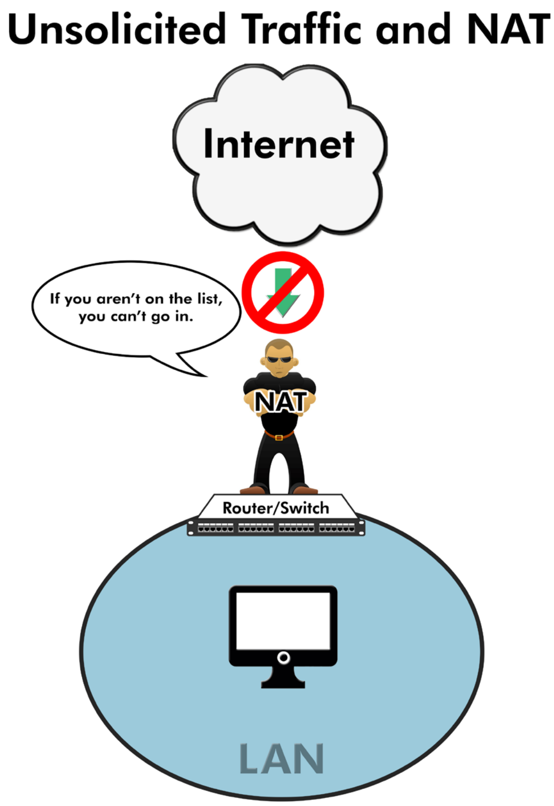
The router will now wait for the response from the web site (it's smart enough to know to expect something from the destination address of the packet it sent). It will then change the destination address of the packet to the private IP address of the laptop before sending it along to the LAN.



In reality, NAT is more complex than this, changing port numbers as well, but we've kept the concept to the bare basics to outline why NAT hurts VoIP.

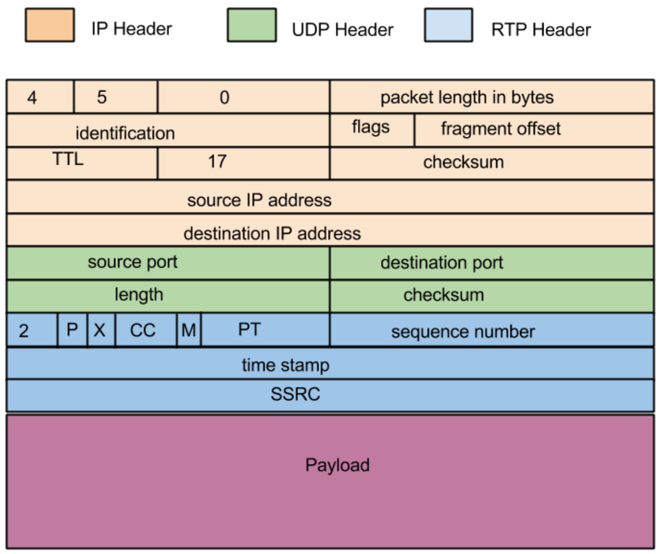
NAT provides for many benefits, including address reuse and basic security. This security exists because packets that arrive from the public Internet without being requested from within the LAN will be discarded. But it's this security element that makes VoIP difficult when using NAT. The concept of placing a VoIP call to a device behind a NAT requires that the NAT deliver unsolicited packets from the Internet to the VoIP device.

This is a complex topic, and as we'll see later on, NAT traversal can cause all sorts of trouble for VoIP.



## Real Time Protocol

A fundamental building block of VoIP is the Real-Time Protocol (RTP). This is a protocol layer that exists within a UDP packet specifically designed to transfer audio (and video) media with low delay. RTP consists of a header that is applied directly after the UDP header in the packet, followed by a media "payload" which consists of the actual encoded audio of a VoIP call.

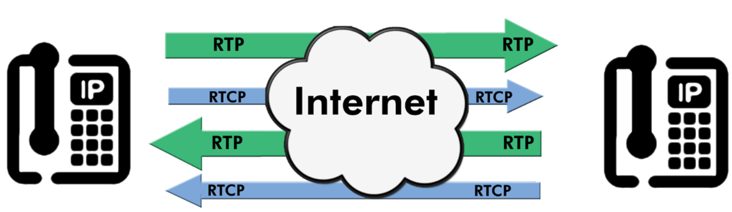



The primary responsibility of the information in the RTP header is to allow the decoder to find the proper playout sequence of the media contained in the packet. RTP doesn't contain any intelligence about what is actually contained in the payload--this has to be handled by other means.

An RTP stream is unidirectional. If a duplex stream is required, an additional independent RTP stream must be initiated in the reverse direction (This function is handled by the Session Initialization Protocol (SIP) layer discussed later).

Finally, an RTP stream (or session, as it's called) has a companion stream that is initiated and travels alongside it for the duration of its life. It's called RTCP and is sent to the same IP address as the RTP stream, but at one port higher. It's used for RTP stream quality statistics but doesn't carry any actual audio, so it uses a small amount of data. But it's important to know about if you're troubleshooting firewall or NAT issues.

**RTP Diagram**



RTP alone can be the basis of a very primitive VoIP call. If each end of the call knows in advance information about encoders used, no NAT routers are involved, and the call can be manually initiated and answered on each end, RTP streams can be "pushed" between the destinations and will provide the path for VoIP. Of course, real-world VoIP involves much more, so we need to add complexity to the system.

## Encoders

Broadcasters who've used POTS, ISDN or IP audio products are familiar with the concept of encoding compression. This is the choice of encoder within the system used to compress digital audio so it uses less network capacity. Encoders like MP3 and AAC are common in that world.

You'll see the VoIP industry use the term "codecs" for this function. But because broadcast transmission devices are also termed "codecs", we'll reserve it to describe hardware, and use "encoders" to describe compression algorithms.

VoIP has its own spectrum of useful encoder choices. VoIP encoders require very low delay and reasonable computational complexity. The RTP protocol has definitions for how to fit all popular encoder payloads into a session.

### G.711

The lowest common denominator encoder in VoIP is the same one that has been used by digital telephone networks for decades, defined as G.711. It's a simple way to compress audio, resulting in a network utilization of 64 Kb/s per channel in each direction, a compression of about only 30% from the original uncompressed stream. This is considered the highest amount of allowable data for a single call by modern standards, and it can add up quickly as multiple calls are handled on the same network. To its benefit, the encoder requires virtually no computer power to compress or decompress.

G.711 is limited in terms of audio fidelity by the choice of its audio sampling rate. Calls using this encoder usually provide only 300 Hz-3 KHz audio response, resulting in the familiar thin sound of phone call, especially when put "on the air".

G.711 actually has two variants, one used mostly in North America ($\mu$-law), and another used elsewhere (a-law). These are defined by the names of the tables used within the encoders to compress. All Comrex codecs and VoIP devices support G.711.

### G.729a

Because G.711 is a bit old and primitive, an encoder has been developed to deliver equivalent audio quality while using a fraction of the network bandwidth. G.729a implements a more aggressive compression algorithm, resulting in network usage of around 8 Kb/s per channel, or about 1/8th the data of G.711. This can be very helpful for avoiding excessive network congestion. Of course, equivalent audio means the same limited fidelity as G.711.

This encoder is sometimes simply referred to as G.729 (without the a), but is equivalent to the user. Another variant, G.729ab, is sometimes available that can detect when voice is present and squelch the data stream during periods of silence, further conserving network bandwidth. Comrex STAC VIP supports G.729a.

### G.722

Familiar to ISDN broadcasters, G.722 is an encoder designed to increase the audio fidelity of phone calls. Using the same network bandwidth as G.711 (64 Kb/s each way), G.722 more than doubles the audio spectrum conveyed by the call, making the caller sound much more natural and identifiable. The 7 KHz spectrum carried by G.722 covers the majority of human voice energy, excluding only the most sibilant sounds in speech.

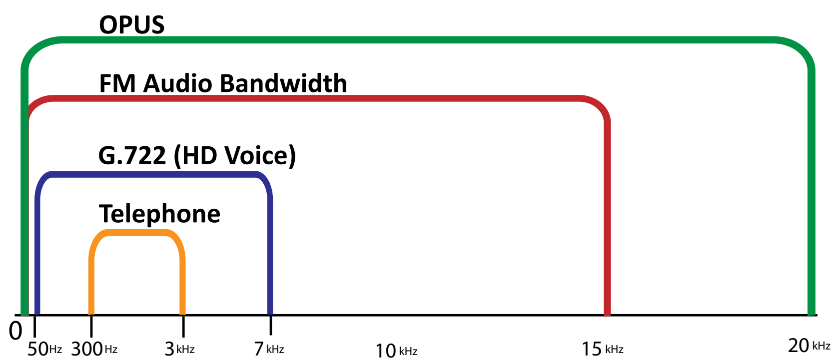
G.722 is the most common encoder for calls that are classified as "HD Voice" in the VoIP world. All Comrex codecs and VoIP devices support G.722.

### Opus

Efforts are increasing at combining the worlds of VoIP and web services. Many web audio services have standardized on Opus, an encoder that delivers near-CD quality audio with low delay. As these efforts continue, users can expect to find more support for the Opus codec in VoIP devices and networks. All Comrex codecs and the STAC VIP phone system support Opus.

### Other encoders

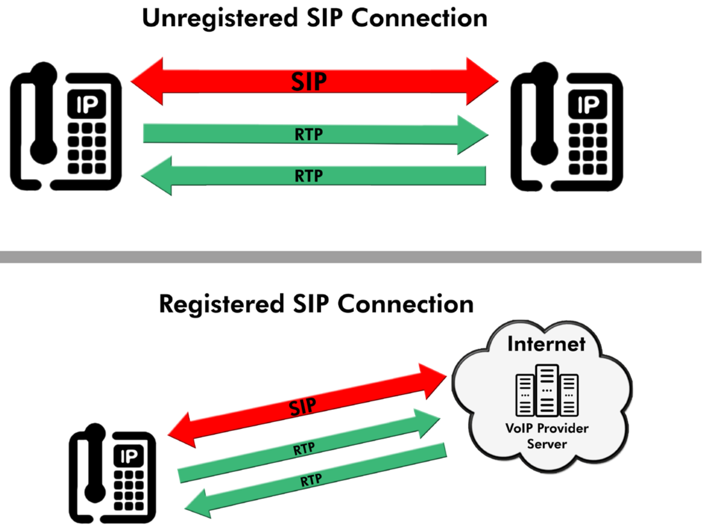
A large spectrum of VoIP-ready encoders have been introduced in the past decades, each having proponents and particular advantages for certain applications. These include iLBC, iSAC, G.722.1, G.722.2, G.726, VMR-WB, SILK and AMR-WB+. For the most part, we expect the industry to support only the four encoders outlined above in most equipment and networks.



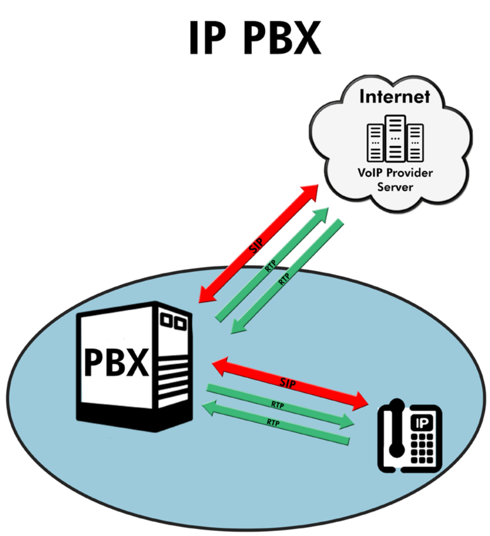

## Session Initialization Protocol

The piece that ties RTP sessions and encoders together, and gives VoIP its telephone-like qualities, is another completely separate connection between devices called the SIP. You'll see the term SIP thrown around in place of VoIP in many places (SIP Phones, SIP PBXs). It's a very powerful specification and is being used for an increasing number of applications besides VoIP, like compatibility standards between broadcast IP hardware codecs, studio-style AoIP installations, and real-time web audio and video. It's becoming such a vital element of so much new technology, it's a very valuable thing to be expert in.

SIP connections can be made in two primary ways--registered and unregistered. In unregistered mode, a SIP channel is opened between devices at the time a call is placed. In registered mode, a SIP channel is constantly maintained between a SIP client (like a studio talkshow system) and a SIP server (like that at an Internet Telephone Provider). Most VoIP users will only use registered mode, so that's what we'll focus on going forward.

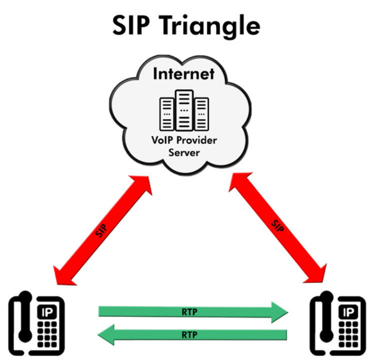



The SIP protocol can be used in more than one link in a VoIP chain. The best example would be a purely IP PBX. In this case, the PBX maintains a SIP channel to an Internet Telephone provider on its WAN port. It also maintains several SIP connections over its LAN to telephone extensions. Because the protocol used in these links is identical, it provides for a lot of flexibility. For example, if need be, the telephone extensions could register directly with the provider, bypassing the PBX entirely.

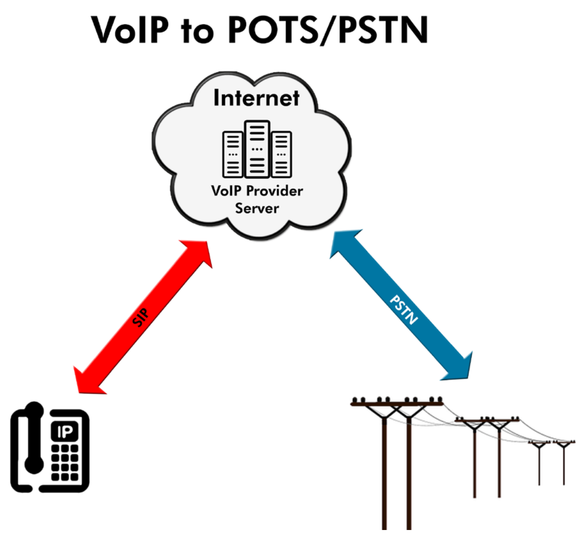



It's important to understand that the SIP protocol does not carry any actual voice between devices-- it simply instructs devices to create separate RTP sessions in each direction. RTP streams are created and destroyed based on commands contained in SIP messages when calls are made or received.

Sometimes the SIP channel is connected to a server that is removed from the RTP sessions entirely. This would likely be the case when two SIP devices are registered to the same (or sometimes even different) providers. The SIP channel would instruct the devices to create RTP sessions between them, rather than to the provider. This is known as the "SIP Triangle".




But more commonly, a SIP device is interested in making and receiving calls to and from the "old fashioned" public switch telephone (PSTN) or "plain old telephone" (POTS) network, whether wired or cellular. In this case both the SIP channel and the RTP sessions are made to a server at the Internet Telephone Provider, and the provider acts as a gateway for the voice call to the "legacy network". The user would be delivered a "real" phone number (DID for Direct Inward Dial) and the provider would handle all the necessary VoIP <-> PSTN conversions. We'll focus on this scenario from here on.



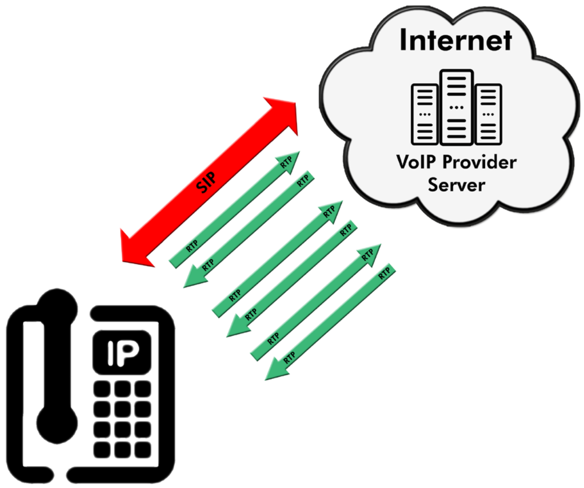

## SIP Details

The technical details of SIP are widely available on the web for further research. But essentially, commands and formats are provided to invite users to a call, accept calls, end them, and reject them. SIP also provides a mechanism to register and authenticate with a server.

Another useful function in SIP is encoder negotiation. The SIP protocol can inform users of which encoders are supported on each end of a session and in which priority. In this way, it's easy to make decisions about which encoder to choose that will be in common with both ends, and to reject calls if no common encoder is found.

Like RTP sessions, the SIP channel utilizes the UDP protocol by default. There is a specific port defined, 5060, as the default "well-known" port over which SIP operates, although it can usually be configured to be different.

A single SIP channel can manage multiple RTP sessions simultaneously. In this way, only a single account needs to be registered with the Internet Telephone Provider and a single SIP channel maintained, but multiple VoIP calls can be run simultaneously. Whenever a call is initiated or dropped, a pair of RTP sessions is created or destroyed on the fly for each call.




## Challenges with SIP/RTP

To summarize the previous sections, most VoIP connections involve a continuously active SIP channel initiated from the user device to a service provider over port UDP 5060. Using this channel, the two ends negotiate calls and create and destroy RTP sessions (each consisting of one RTP and one RTCP) in each direction. Like the SIP channel, these sessions also run between the end-user and provider, so the provider can bridge them to the legacy phone network. The SIP channel also negotiates which encoders will be used on the RTP channels.

So what can possibly go wrong? Almost every issue can be run down to NAT-based routers or blocking firewalls.

### Issues with the SIP channel

The SIP channel generally has the fewest issues, since it's usually originated from the user end of the link. This means NAT routers on the user end will generally allow this outgoing traffic to pass, and allow the response traffic (from the provider) back in. But if a network is heavily firewalled in a way that blocks outgoing access to UDP 5060, this channel will never be created and the user cannot register with the provider.

Also, although we have described the SIP connection as "always active", there are periods of inactivity on the link when no calls are being set up or ended. In order to receive information about new incoming calls from the provider, the user end must keep the SIP connection (or "binding") open through the NAT router to prevent it from terminating the binding and blocking incoming traffic. It does this by sending periodic updates even when no changes are being made to any calls. The interval of these updates is usually adjustable, but must be shorter than the timeout value the router takes to shut down any unused bindings.

### Where am I?

According to the SIP standard, the user device will inform the provider of its IP address (over the SIP signaling connection), and the provider will "push" the RTP session containing the incoming voice to that address. But devices on LANs often don't know what their "public" address is, only the private one assigned to them on the LAN. If the provider tries to initiate a stream to that address, it will go nowhere.

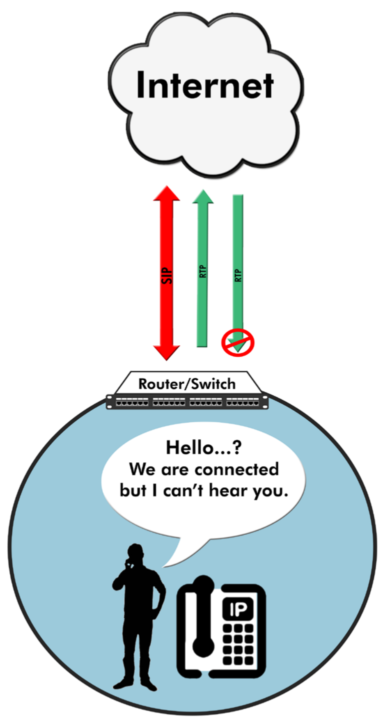
Many VoIP providers install a "cheat" here that will look at the user's IP address and determine if it looks "private". If so, they will ignore it and send the RTP stream to the destination address of the RTP session they receive.

If the cheat isn't implemented, user devices have a way of looking up their public IP address via a protocol called STUN. This protocol can usually be enabled within the user's equipment configuration. If enabled, the device will look to a STUN server out on the public Internet, and query its own address. It will then use that public address to populate the "from" field in the SIP handshake.

### Don't block me, bro!

Even if the provider gets the correct IP address of the user, there's plenty that can go wrong. Remember, SIP involves creating extra RTP "channels" in each direction to carry the actual voice. The ports used on each end are negotiated over the SIP signaling channel for each call. There aren't any "standard well-known" ports used for these connections. And there can be many of them active on different ports if lots of simultaneous calls are happening.

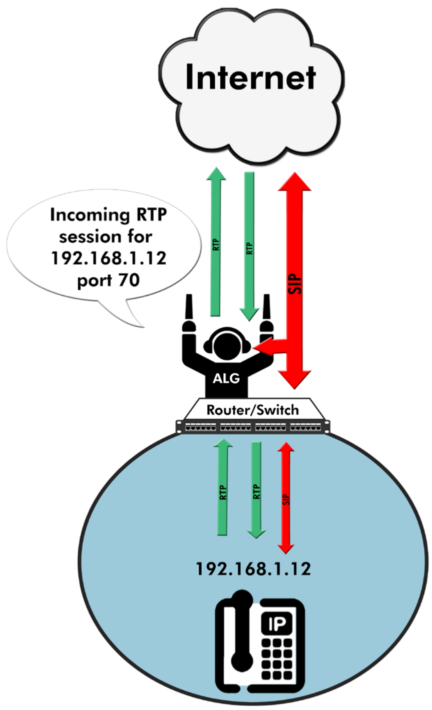
As far as the user's router or firewall is concerned, a new RTP session is trying to make it through its security layer. It's not aware this session has been requested, so it's blocked by default. This usually results in a one-way connection, where no audio can be heard on the SIP user end of the call.



### ALG to the rescue

This scenario has become common enough that router and firewall manufacturers have started to address it. The solution is called SIP ALG (for application layer gateway) and has been built into the firmware of most modern devices. It may be on or off by default. And the quality of how it functions may vary--early implementations sometimes did more harm than good.

But a properly functioning ALG will listen to your SIP channel, and gain an understanding of which RTP sessions are being created on which ports. It will then allow the incoming session through.




In reality, an ALG may often take quite a bit of license with your SIP connection. It can rewrite many of the SIP fields in order to comply with its rules, so the IP and port information getting to the service provider may actually be completely different than those sent by the device. As long as it has the intelligence to open the proper ports, this will usually work fine.

It's even possible that your SIP connection is being processed by more than one ALG, as in the instance of a separate router and firewall on the connection. Of course in this scenario, the possibilities for errors compound. Sometimes it's best to disable unnecessary ALGs in the link. Unfortunately, diagnosing these issues require analyzing packet captures. Luckily, SIP is a well-known protocol that can be easily deciphered by packet capture systems.

## Summary

The important elements of SIP are as follows:

1. An independent connection stays open on UDP 5060 between the user and the service provider
2. Separate and multiple RTP sessions are established in each direction for calls
3. Routers and firewalls interfere with these RTP sessions by design, but ALGs built into these devices can help.

## PBXs

So far we've discussed SIP connections to outside or "cloud" VoIP providers. But many times, the user already has a SIP PBX on premises, which already connects to the public telephone network by VoIP or legacy means, like analog lines or T1s. Since most modern PBXs talk SIP to their extensions, they just need to tie a SIP-compatible device (like a codec or hybrid) to the PBX, and allow the PBX to decide how to route calls to the device.

As mentioned before, the SIP protocol used in this scenario is the same. The device will register and maintain a SIP connection to the PBX, and the PBX will inform the device of incoming calls. RTP channels will be created when required between the SIP device and the PBX. This will usually be successful, since the LAN environment is less reliant on routers, subnets and firewalls to block the RTP channels.

## Registering with a SIP Server or PBX

The process of registering a device to a SIP provider, whether it's in the "cloud" or at your location, is usually simple. Much like registering an email client with a mail server, the VoIP client (the VoIP hardware) must know the location of the server, and a username/password combo with which to register. The server location can be in the form of an IP address, or a URL.

Some servers with more complex arrangements may require more information to help choose options. There may be separate settings for your SIP Proxy server, your SIP domain, and your SIP registration server. There may be choices for encoder support, auth username (an additional credential used for authentication), and caller ID options. For the most part, any essential info that needs to be programmed will be delivered from your provider (or in the case of a PBX, your Telco department) and you can set your VoIP device with the parameters that match, and ignore the others.

## Making and Receiving calls

Once registered correctly with a SIP server, incoming calls will be routed to your SIP device based on the calling plan set up with your provider or PBX. Whether it's the DID line(s) assigned to you by the provider, or an incoming trunk attached to your PBX, a "ring" on the line will trigger the server to notify your device of a call request using the SIP protocol. Your device can accept or reject the call. If you accept the call, an RTP channel is created to your device each way.

Outgoing calls just reverse the process. The SIP device sends an outgoing call request to the server, which attempts to complete the call. Call progress messages will be sent to your SIP device from the server, which may translate them to familiar tones like ringing and busy. On call completion, the server will create the RTP channels in the same way as for incoming calls.

## Hunting

Of particular interest to broadcasters who take lots of calls simultaneously is hunting behavior, or the way the system behaves toward simultaneous incoming calls. Keep in mind, when an incoming call is in the "ringing" state, there are only status messages exchanged over the SIP connection--no actual audio is being transferred. The RTP audio channels are only created after the call is answered.

Only one SIP connection needs be open for multiple voice channels to be created. Your VoIP provider or PBX will be programmed to allow a designated number of simultaneous voice channels, and any further incoming calls will be rejected there. By default, most multi-channel VoIP gear will "hunt" any second, third etc. call to the next "line" on the device. In this way hunting is inherent. If more than the supported number of calls is requested to the VoIP device, it will reject them in the same way as the provider does, and no RTP channel will open for these excess calls.

Alternately, it's possible to set up a separate SIP account for each "line" on the SIP device, and this account should be capable of creating only one "channel" at a time. In this case, it's the responsibility of the provider or PBX to sort the hunting arrangement and notify the proper account about incoming calls.

## Choke Lines

Another topic of interest to broadcasters is choke lines, the specially conditioned telephone trunks designed not to fail under loads of thousands of incoming calls (e.g. for contests). In the PBX scenario, choke lines can easily be used as the trunks that feed the PBX, and very little changes.

When using a cloud provider, it's important to notify them about potential peak call volume to avoid overloading their systems. But cloud providers are usually equipped to provide service to high-volume nationwide call centers, so they can usually implement techniques to throttle large amounts of calls without impacting overall service.

# XII. INFORMATION FOR IT MANAGERS ABOUT VH2

VH2 is an embedded Linux-based device with dual 10/100/1000Base-T Ethernet ports. The device contains an optimized version of the **Linux kernel** (at this writing, **3.12**). The IP parameters are set using an internal web server that hosts a configuration page on **TCP 80**.

Alternately, during the first five minutes of power up, the IP parameters may be set by a PC on the local LAN using a proprietary broadcast UDP protocol. Comrex provides the **Device Manager** software to perform this function on the local PC. After five minutes of operation, this function is disabled.

The device runs several services on different ports, outlined here:

## INCOMING SERVICES

The device hosts a combined HTTP/XML service on **TCP 80**. If this service is needed outside the firewall, the port will need to be routed to the VH2.

Firmware updates to the device are installed using the **Device Manager**. This update process is password protected and done via XML over **TCP port 8080**. In addition to the password protection, the update data itself must have a valid cryptographic signature from Comrex, or else it is rejected. In order for the unit to be remotely updated, **TCP port 8080** must be forwarded to the device. Alternately, updates can be initiated from any local PC using the **Device Manager**.

The device can support connection to a SIP trunking service, which would require incoming service on a single **UDP SIP port** (usually **5060**) and two **UDP RTP ports** in the range of **16384-16432**.

The device can support connection to a registered SIP service. In this case, the **UDP** SIP connection will be outgoing and the two **UDP RTP ports** will be incoming in the range of **16384-16482**.

Typically, SIP services rely on the presence of a SIP ALG within the firewall to open RTP ports.

If Comrex support is required, we may ask for access to the SSH host on the VH2 on **TCP 22**. SSH service is protected by a private keypair which is not delivered to customers. SSH service can be disabled in the setup menu.

## OUTGOING SERVICES

As described above, the VH2 will make outgoing connections to register with SIP providers (usually to port **UDP 5060**) combined with incoming and outgoing RTP in the range of **UDP 16384-16432**.

An NTP client is implemented by default to **UDP 123**.

**Comrex Corporation**
**19 Pine Road**
**Devens, MA 01434**
**USA**

Technical Support is available Monday-Friday 8:30AM-5PM EST.

**1-800-237-1776 (North America)**
**1-978-784-1776 (International)**
**1-978-784-1717 (FAX)**

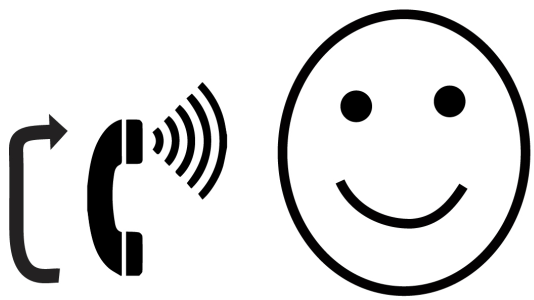
**email techies@comrex.com**

**Product manuals and firmware updates available on the web at:**
**http://www.comrex.com**

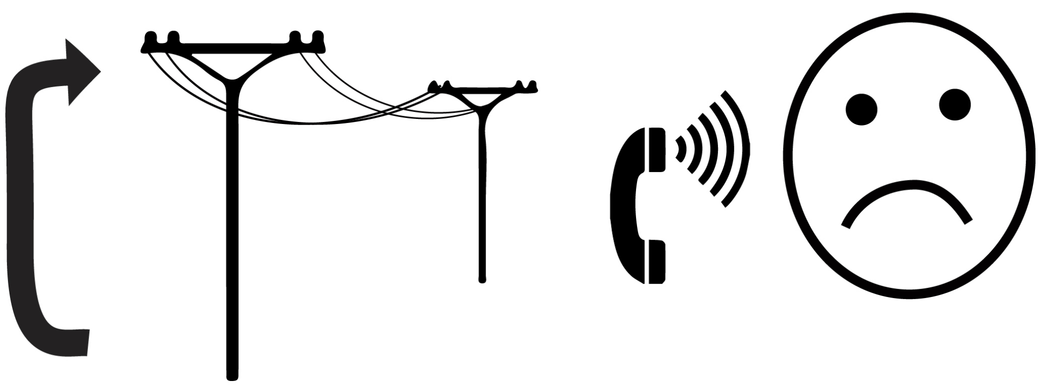
# XIV. WHY DO I HEAR HEAR MYSELF MYSELF?

## MIX-MINUS AND ELIMINATING ECHO

Studio telephone integration is a two-way process. The caller must send his audio to the studio, but also receive a return feed that allows him to interact with other sources, like a host. An important element of voice telephony involves allowing a speaking party to hear his own voice in his own earpiece.  This sidetone provides a speaker the comfort of knowing his voice is getting through, and makes two-way communication flow more easily.



Sidetone created within a telephone has no delay
and makes conversation easier



Sidetone created on the far end of a call has delay
and makes conversation difficult

But for several reasons, telephone sidetone is always generated locally within the speaker's equipment, rather than on the far end of the call. This is because humans have a very hard time handling even the smallest delay in this sidetone signal. In testing, we find that any delay over around 10mS starts to have an effect called “slapback” where the speaker is unable to maintain conversation and begins to halt and stutter.

Even in “old-fashioned” analog telephone circuits, it’s possible to create a 10mS round-trip delay on a long distance call. Now add in the requirement that modern VoIP-based systems have inherent windowing and buffer delays, and its easy to pile up over 100mS round-trip on a call. A delay of this length will typically not impede interactive conversation, but will certainly create an intolerable “slapback” environment if the caller hears his own voice delayed.

The telephone network employs digital echo cancellers at various nodes along the path of a phone call to avoid this scenario. And when they malfunction or are “untrained” at the start of a call, the effect is a dramatic echo in the caller’s ear.

Many users installing a studio-based phone system for the first time make the mistake of applying audio to the outgoing “send’ port that contains the main program feed - the same audio used to feed the transmitter or webstream. Since this mix contains the caller’s own audio, and there’s an inherent delay in modern digital systems, the “slapback” effect is immediate.

The solution here is mix-minus-- a term used for a special mix of audio that explicitly excludes one source--the audio coming from the place the mix-minus is being sent. To put it another way, mix-minus is the entire studio mix minus one audio source.

So how do we create this special audio mix? On modern studio systems, this is usually well defined and easy to do. Many consoles feature channels dedicated to telephone interface, and part of the channel is an automatically-created mix-minus output.

In less full-featured consoles, a mix-minus can often be created with an auxiliary or “audition” bus function. By selecting all relevant incoming sources on the bus except for the telephone fader, you can do this easily. Figure 10 shows the block diagram of a single mix-minus feed being generated on a mixing console.

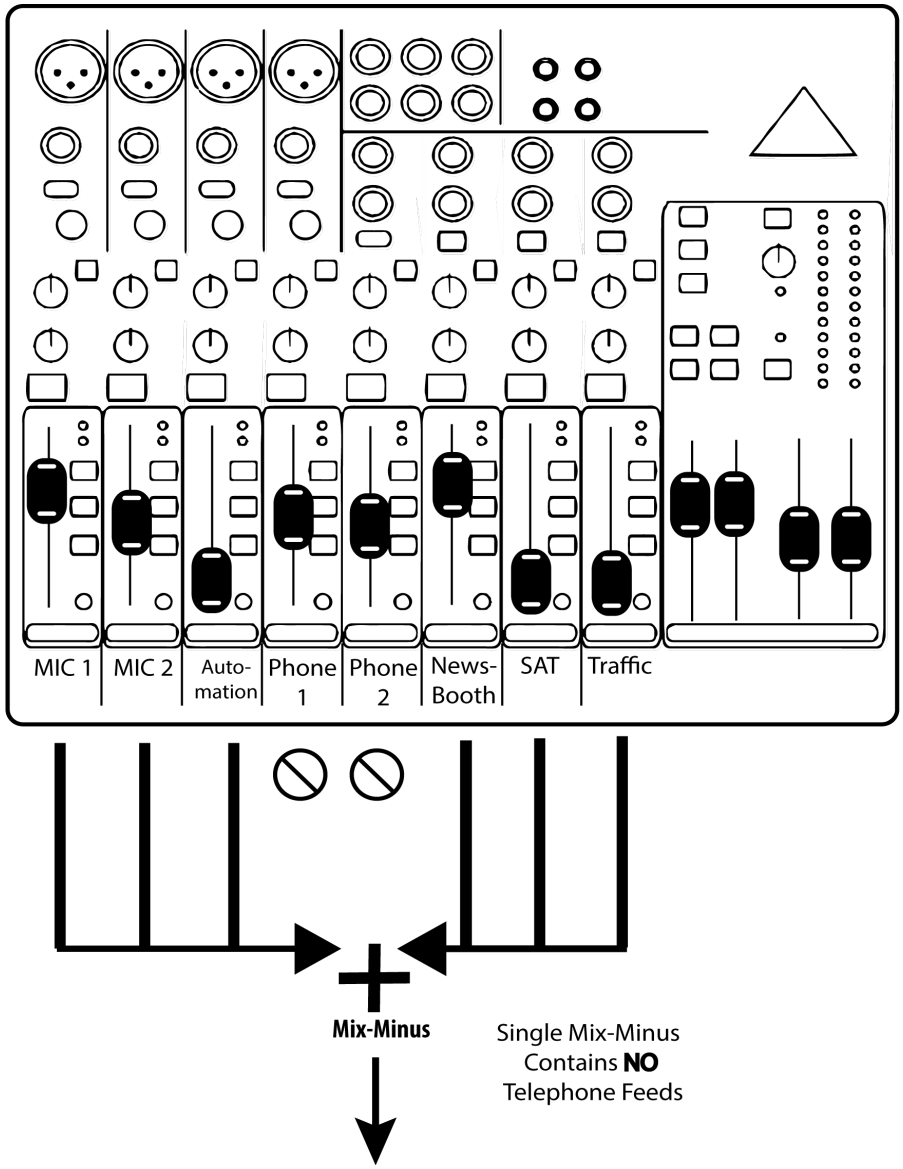



FIGURE 10 **SINGLE MIX-MINUS NO TELEPHONE FEEDS**

In some environments, it's only important that the caller hear the in-studio host, and less urgent that the on-air caller be able to hear automation, news reports, codecs etc. It's possible that simply routing an amplified version of the studio microphone signal to the "send" input will meet those needs.

Some studio telephone systems, like Comrex VH2, allow telephone callers to appear on one of two outputs (and therefore on two, separate console faders). In this circumstance, you often have a choice of delivering a single mix-minus with neither of the telephone audio sources present, or two distinct mix-minus feeds. In the case of two feeds, it's important to note that mix-minus A must include the caller audio B and vice-versa. Figure 10 illustrates this concept. Many consoles with dual telephone channels are designed to work this way, and should deliver the correct set of mixes automatically.

Finally, if you're stuck in a studio where you need to create multiple mix-minus feeds with no resources to do so, investigate the Comrex **Mix-Minus Bridge** product. It's an easy way to deliver up to six distinct mix-minus feeds (e.g. for two telephone channels, two audio codecs, a two-way radio system and an RPU return channel) simultaneously while sacrificing only one console auxiliary bus.

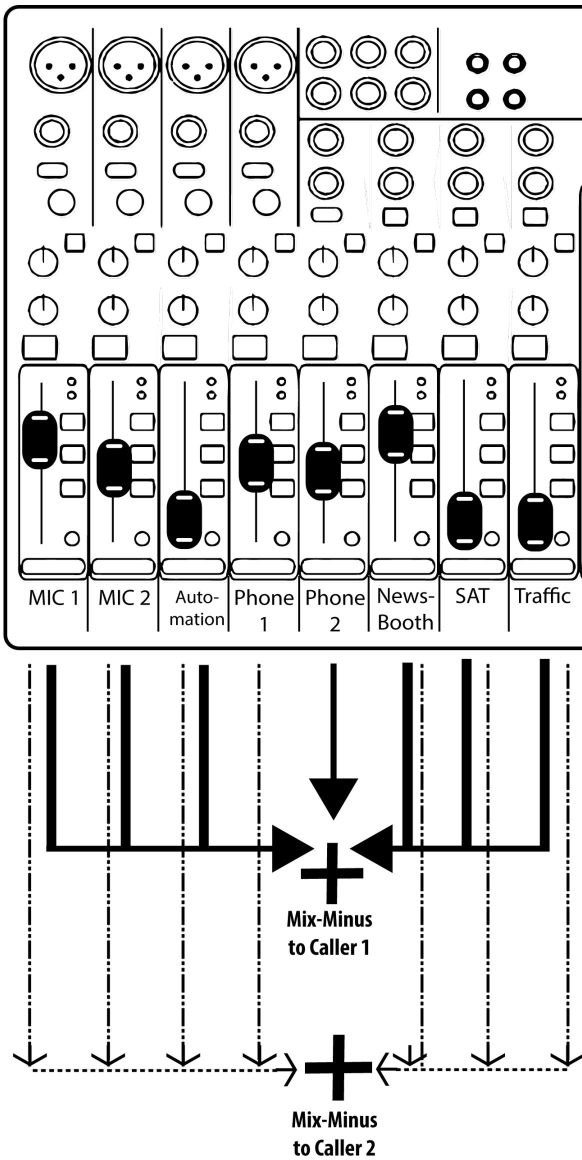



FIGURE 11 **TWO MIX-MINUS FEEDS**

# xv. SOFTWARE LICENSES

VH2 contains both proprietary and open source software. All proprietary portions are 

COMREX CORPORATION

END USER LICENSE AGREEMENT

**IMPORTANT READ CAREFULLY: This End User License Agreement (Agreement) is a legal and binding agreement between you (either an individual or a single entity) and Comrex Corporation (Comrex) for the software contained within this product.**

**WHEN YOU INSTALL AND USE THIS PRODUCT, YOU ARE CONSENTING TO BE BOUND BY AND ARE BECOMING A PARTY TO THIS AGREEMENT. IF YOU DO NOT AGREE TO (OR CANNOT COMPLY WITH) ALL OF THE TERMS OF THIS AGREEMENT, DO NOT INSTALL OR USE THIS PRODUCT. APPROVAL IS NOT A PREREQUISITE TO THE VALIDITY OR ENFORCEABILITY OF THIS AGREEMENT AND NO SOLICITATION OF ANY SUCH WRITTEN APPROVAL BY OR ON BEHALF OF COMREX SHALL BE CONSTRUED AS AN INFERENCE TO THE CONTRARY. IF THESE TERMS ARE CONSIDERED AN OFFER BY COMREX, YOUR ACCEPTANCE IS EXPRESSLY LIMITED TO THESE TERMS.**

**Under this Agreement, Comrex offers you use of the proprietary Comrex software contained in this product under the following terms:**

1. **No Warranty.**

WITH THE EXCEPTION OF THE GENERAL PRODUCT WARRANTY OFFERED AT THE BEGINNING OF THIS MANUAL COMREX EXPRESSLY DISCLAIMS ANY OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT.

FURTHER, COMREX DOES NOT WARRANT RESULTS OF USE OR THAT THE SOFTWARE IS BUG FREE OR THAT ITS USE WILL BE UNINTERRUPTED. COMREX AND ITS LICENSORS DO NOT AND CANNOT WARRANT THE RESULTS YOU MAY OBTAIN BY USING THE SOFTWARE. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS AGREEMENT.

2. **Limitation of Liability.**

NOTWITHSTANDING ANYTHING ELSE HEREIN, UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, INCLUDING, BUT NOT LIMITED TO, TORT, CONTRACT, NEGLIGENCE, STRICT LIABILITY, OR OTHERWISE, SHALL COMREX OR ITS LICENSORS BE LIABLE TO YOU OR ANY OTHER PERSON (I) FOR ANY PUNITIVE, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, ACCURACY OF RESULTS, COMPUTER FAILURE OR MALFUNCTION, OR DAMAGES RESULTING FROM YOUR USE OF THE SOFTWARE OR THE SERVICES PROVIDED BY OR ON BEHALF OF COMREX, OR (II) FOR ANY MATTER BEYOND COMREXS OR ITS LICENSORS REASONABLE CONTROL. COMREXS AND ITS LICENSORS MAXIMUM LIABILITY FOR DAMAGES OF ANY KIND WHATSOEVER ARISING OUT OF THIS AGREEMENT SHALL BE LIMITED TO THE PURCHASE PRICE PAID BY YOU FOR THE SOFTWARE, EXCEPT WHERE NOT PERMITTED BY APPLICABLE LAW, IN WHICH CASE COMREXS LIABILITY SHALL BE LIMITED TO THE MINIMUM AMOUNT PERMITTED BY SUCH APPLICABLE LAW. THE FOREGOING LIMITATIONS SHALL APPLY EVEN IF COMREX HAS BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES.

3. **Indemnity.**

You shall indemnify and hold harmless Comrex and its licensors from any third party claims, damages, liabilities, costs and fees (including reasonable attorney fees) arising from Your use of the Software as well as from Your failure to comply with any term of this Agreement.

4. **General.**

This Agreement represents the complete agreement concerning this license between the parties and supersedes all prior agreements and representations between them. This Agreement may be amended only by a writing executed by both parties. If any provision of this Agreement is held to be illegal or unenforceable, that provision shall be limited or eliminated to the minimum extent necessary to make it legal and enforceable and this Agreement shall otherwise remain in full force and effect and enforceable. The failure of Comrex to act with respect to a breach of this Agreement by You or others does not constitute a waiver and shall not limit Comrexs rights with respect to such breach or any subsequent breaches. This Agreement shall be governed by and construed under Massachusetts law (without regard to its conflicts of laws provisions) as such law applies to agreements between Massachusetts residents entered into and to be performed within Massachusetts. The sole and exclusive jurisdiction and venue for actions arising under this Agreement shall be the State and Federal courts in Worcester County, Massachusetts; You hereby agree to service of process in accordance with the rules of such courts. The party prevailing in any dispute under this Agreement shall be entitled to its costs and legal fees.

Questions concerning this Agreement should be sent to the address set forth below. Any notices or correspondences will only be effective if sent to such address.

**Comrex Corporation**
**19 Pine Road**
**Devens, MA 01434**

Open source software used in VH2 is protected under a wide variety of licenses, listed below.

## GNU PUBLIC LICENSE VERSION 2

linux
module-init-tools
udev
e2fsprogs
busybox
bash
tcpdump
alsa-utils
ethtool
acpid
usbutils
procps
fxload

### GNU GENERAL PUBLIC LICENSE



**Preamble**

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

## GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The “Program”, below, refers to any such program or work, and a “work based on the Program” means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term “modification”.) Each licensee is addressed as “you”.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program’s source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

**NO WARRANTY**

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF

MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

**END OF TERMS AND CONDITIONS**

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the program's name and an idea of what it does.
Copyright (C) yyyy  name of author

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details
type `show w'.  This is free software, and you are welcome
to redistribute it under certain conditions; type `show c'
for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright
interest in the program `Gnomovision'
(which makes passes at compilers) written
by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

## GNU PUBLIC LICENSE VERSION 3

m4
gnupg
dosfstools



**Preamble**

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and
(2) offer you this License giving you legal permission to copy, distribute and/or modify it.
For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS**

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users’ Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work’s users, your or third parties’ legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program’s source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:
•a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
•b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
•c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
•d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:
•a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
•b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
•c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
•d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
•e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed.

Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:
•a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
•b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
•c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
•d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
•e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
•f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others’ Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

**END OF TERMS AND CONDITIONS**

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program.  If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:
<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

## LESSER GNU PUBLIC LICENSE VERSION 2.1

sysfsutils
qt

### GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999
Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

**Preamble**

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the “Lesser” General Public License because it does Less to protect the user’s freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users’ freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a “work based on the library” and a “work that uses the library”. The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called “this License”). Each licensee is addressed as “you”.

A “library” means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The “Library”, below, refers to any such software library or work which has been distributed under these terms. A “work based on the Library” means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term “modification”.)

“Source code” for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a “work that uses the Library”. Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a “work that uses the Library” with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a “work that uses the library”. The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a “work that uses the Library” uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a “work that uses the Library” with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer’s own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

> a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable “work that uses the Library”, as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

**NO WARRANTY**

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

**END OF TERMS AND CONDITIONS**

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
one line to give the library's name and an idea of what it does.
Copyright (C) year name of author

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names: Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written
by James Random Hacker.

signature of Ty Coon, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

## MOZILLA PUBLIC LICENSE VERSION 1.1

freeswitch

1. Definitions.

1.0.1. “Commercial Use” means distribution or otherwise making the Covered Code available to a third party.

1.1. “Contributor” means each entity that creates or contributes to the creation of Modifications.

1.2. “Contributor Version” means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. “Covered Code” means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. “Electronic Distribution Mechanism” means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. “Executable” means Covered Code in any form other than Source Code.

1.6. “Initial Developer” means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. “Larger Work” means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. “License” means this document.

1.8.1. “Licensable” means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein. 1.9. “Modifications” means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

a.Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

b.Any new file that contains any part of the Original Code or previous Modifications.

1.10. “Original Code” means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. “Patent Claims” means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. “Source Code” means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor’s choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. “You” (or “Your”) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, “You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means

(a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or

(b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

a.under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

b.under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

c.the licenses granted in this Section 2.1 (a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

d.Notwithstanding Section 2.1 (b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

a.under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

b.under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

c.the licenses granted in Sections 2.2 (a) and 2.2 (b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

d.Notwithstanding Section 2.2 (b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the legal file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4 (a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Sections 3.1, 3.2, 3.3, 3.4 and 3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the legal file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. Disclaimer of warranty

Covered code is provided under this license on an “as is” basis, without warranty of any kind, either expressed or implied, including, without limitation, warranties that the covered code is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the covered code is with you. Should any covered code prove defective in any respect, you (not the initial developer or any other contributor) assume the cost of any necessary servicing, repair or correction. This disclaimer of warranty constitutes an essential part of this license. No use of any covered code is authorized hereunder except under this disclaimer.

8. Termination

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as “Participant”) alleging that:

a.such Participant’s Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

b.any software, hardware, or device, other than such Participant’s Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant’s Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. Limitation of liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall you, the initial developer, any other contributor, or any distributor of covered code, or any supplier of any of such parties, be liable to any person for any indirect, special, incidental, or consequential damages of any character including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to you.

10. U.S. government end users

The Covered Code is a “commercial item,” as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of “commercial computer software” and “commercial computer software documentation,” as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. Miscellaneous

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. Responsibility for claims

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. Multiple-licensed code

Initial Developer may designate portions of the Covered Code as “Multiple-Licensed”. “Multiple-Licensed” means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the MPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

Exhibit A - Mozilla Public License.

“The contents of this file are subject to the Mozilla Public License Version 1.1 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an “AS IS” basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is ______________________________________.

The Initial Developer of the Original Code is ________________________.
Portions created by ______________________ are Copyright (C) _____________________________. All Rights Reserved.

Contributor(s): ______________________________________.

Alternatively, the contents of this file may be used under the terms of the _____ license (the “[___] License”), in which case the provisions of [______] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License.”

NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.

## Dropbear

Dropbear contains a number of components from different sources, hence there are a few licenses and authors involved. All licenses are fairly non-restrictive.

The majority of code is written by Matt Johnston, under the license below.

Portions of the client-mode work are (c) 2004 Mihnea Stoenescu, under the same license:

Copyright (c) 2002-2008 Matt Johnston
Portions copyright (c) 2004 Mihnea Stoenescu
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

LibTomCrypt and LibTomMath are written by Tom St Denis, and are Public Domain.

=====

sshpty.c is taken from OpenSSH 3.5p1,
Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland
All rights reserved
“As far as I am concerned, the code I have written for this software can be used freely for any purpose. Any derived versions of this software must be clearly marked as such, and if the derived work is
incompatible with the protocol description in the RFC file, it must be called by a name other than “ssh” or “Secure Shell”. “

=====

loginrec.c

loginrec.h
atomicio.h
atomicio.c
and strlcat() (included in util.c) are from OpenSSH 3.6.1p2, and are licensed
under the 2 point BSD license.

loginrec is written primarily by Andre Lucas, atomicio.c by Theo de Raadt.

strlcat() is (c) Todd C. Miller

=====

Import code in keyimport.c is modified from PuTTY's import.c, licensed as
follows:

PuTTY is copyright 1997-2003 Simon Tatham.

Portions copyright Robert de Bath, Joris van Rantwijk, Delian Delchev, Andreas Schultz, Jeroen Massar, Wez Furlong, Nicolas Barry, Justin Bradford, and CORE SDI S.A.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE
FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**STRACE**

Copyright (c) 1991, 1992 Paul Kranenburg
Copyright (c) 1993 Branko Lankester
Copyright (c) 1993 Ulrich Pegelow
Copyright (c) 1995, 1996 Michael Elizabeth Chastain
Copyright (c) 1993, 1994, 1995, 1996 Rick Sladkey
Copyright (C) 1998-2001 Wichert Akkerman
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.