Ginsu Talk, Version 1.6.9, revision 6
Installation and Setup Guide

Written by: Scott Bachmann
May 2003, Revision 01
Send comments to u550@phringe.org

Contents

Part 4. Setting Up Features

Appendix

Typographical Conventions

Throughout this manual, three different typographical conventions are used. By understanding these conventions, it will be easier to follow the examples that are given later in the text. The three conventions are as follows:

Typeface	Usage	Example
`Courier`	Command names and files within text and for screen output.	Modify the `login.txt` file. Run `gtstart` to start Gtalk.
`Bold Courier`	Text that has been typed and echoed onto the screen (input).	% **`bin/gtstart`**
Italic *Times new roman*	Used for emphasized text, important terms, and variables that should be replaced.	You *must* remember … Run **`./configure`** ***--option*** to configure the software.

Also, a "**. . .**" is used to represent the output of the previous command that is not shown in this documentation.

Shell Prompts

Throughout this manual, two different shell prompts will be shown. By understanding these prompts, it will be easier to follow the examples that are given later in the text. The two prompts are as follows:

Shell type	How to get there	Example Prompt
Bash Shell	**`bash --login`**	[*`user`*] $
Bash Shell as super-user	**`su – root`** **`bash --login`**	[root] #

Part 1. Background

What is it?

Ginsu Talk, also referred to as Gtalk, is a telnet chat software application designed for Linux. It allows people to login as a guest or registered user to communicate with each other using text messages in private or public *channels*. It also incorporates a local mail system and a bulletin board system for message posting.

Why it is?

Some people might ask, "Why develop another chat software?" The answer is that Gtalk wasn't developed recently. Gtalk first arose in 1992, as developed by David Jeske and Dan Marks. It was first designed to be connected to a modem pool and run under MS-DOS©, and later OS/2 Warp©. Unlike the current version, a user would dial-in to the system directly, similar to the old BBS systems of the late 1980's and early 90's. As the BBS scene was replaced with the Internet, Gtalk needed to adjust and was rewritten for Linux with telnet capabilities. This allowed existing Gtalk systems to change with the times, and to continue providing a "local" chat system for their users, while also allowing the system to expand (and retain) it's user database. Since Gtalk began with modems, most users on current Gtalk systems are still local with each other, and friends in real life. Those who are no longer local can still use the system to keep in touch with their friends from their hometown. Gtalk assists in facilitating the users with an easy means of communicating with a social group, while eliminating annoyances of other chat software. Since it is locally administrated, more control can be exerted to prevent disruptions and built-in facilities allow regulation of who can use the system. Also, Gtalk is a self-contained server package, which requires no special client software (as almost all OSes come with a telnet client pre-installed, or a user can use one of their own – CRT tends to be preferred), and is entirely keyboard driven (eliminating the annoying multi-window, mouse and keyboard switching). A number of web-based java telnet clients also exist, which can allow a user to visit a web page and login to the system. The Phringe (http://www.phringe.org) hosts such a site in addition to their basic telnet capabilities.

How does it work?

Gtalk requires a Linux based system in order to function. It runs as a server processes (`gtalk`) listening upon one or two ports (often 23 and 4000). When a user establishes a connection to one of the listening ports, that connection is assigned the next available node number. This node number allows the system to keep track the user, their information, and provide them with private messages sent to their node number. The server passes the connection off to a client process (`gtclient`) that interacts with the user.

The first task the `gtclient` performs is to ask for the user to login. They can either login as a guest, or as a registered user by entering their user number and password. If the user is a registered user, their unique *user* record and *class* information is loaded. The *user* record contains information such as their handle, real life information, statistics, and their *class.* A *class* is a record that contains information about privileges, such as which commands are permitted and what *priority* they are. The SYSOP class is the administrative class. Any user that is associated with this class has permission to run all commands on the system, and has a priority level of zero, the most powerful.

After completing the login process, the user is placed into the main channel. At this point, the user can either type a message, or enter a *command*. A *command* is an instruction entered by the user which begins with a '/' character. Any other text is interpreted as a message that should be sent to the channel the user is in.

Part 2. Setting up the software

Requirements

To install Gtalk, you *must* meet the following requirements:

- Have an operational Linux system, including development packages that contain binutils, gcc, gunzip, and tar, with at least a 2.4 kernel.
- Have at least 12 MB of disk space (more is required for message storage).
- Have at least 32 MB of memory.
- Have super-user access *or* the super-user's assistance during installation.
- Have basic Linux development and shell skills.
- Have downloaded the latest version (by date) from the Yahoo group's site: http://groups.yahoo.com/group/opengtalk.

Configuration

First, you will need to extract the archive using `gunzip` and `tar`:

```
[user]$ gunzip -d -c gtalk-latest.tar.gz | tar -xf -
[user]$ cd gtalk-latest
```

Gtalk's configure script has several options that you can pass to it:

--prefix=/path	The prefix is the location where Gtalk is installed, /home/gtalk by default. This option allows you to override the default and specify an alternate location. We strongly advise that you install Gtalk in an empty directory. The prefix is later referred to in this document as {*GTALK_HOME*}.
--with-modems	This option will enable modem support within Gtalk. This code has been commented out using #ifdef/#endif macros as it is rarely used.
--with-credit	This option will enable accounting support within Gtalk. This code has been commented out using #ifdef/#endif. If enabled, this code allows for a credit-based system, where a user will need credits, given by the administrator(s), to use the system.

Run the configure script now, with any options if needed, to prepare the code.

```
[user]$ ./configure [options]
  ...
```

Building

Once the configuration process has been completed, you'll need to build the Gtalk package. Building the package consists of compiling the 'C' code into a machine-readable format. It also builds the required shell scripts that accompany Gtalk. Use the command *make* to begin compiling:

```
[user]$ make
  ...
```

Installation

During this installation process, you *must* have super-user privileges, as Gtalk will require it to run[†]. If you do not have the root password to the system you wish to install the software upon, speak with your Systems Administrator to have them install it. If you have the root password, run the following command.

```
[user]$ su - root
Password: (root password)
```

Once you've become super-user, run the following command to begin the installation process of the Gtalk binaries, scripts, and support files:

```
[root]# make install
  ...
```

Now run the following commands, also as super-user, to create the necessary accounts and set the file permissions. These commands also appear in the file {*GTALK_HOME*}/info/FINAL-INSTALL.

```
[root]# groupadd gtalk
[root]# useradd -c "Gtalk System" -d "{GTALK_HOME}" -g gtalk -M gtalk
[root]# passwd -l gtalk
[root]# useradd -c "Gtalk Guest" -d "{GTALK_HOME}/users/gtgst" –g \
        gtalk gtgst
[root]# passwd -l gtgst
[root]# useradd -c "Gtalk Sysop" -d "{GTALK_HOME}/users/gtsys" -g \
        gtalk gtsys
[root]# passwd gtsys
  ...
```

[†] It is –hypothetically– possible to run Gtalk without being root. Your setup would require a specially crafted chroot directory and may result in a number of security problems if done improperly. Such a setup is outside of the scope of this documentation.

```
[root]# chown -R gtalk:gtalk {GTALK_HOME}
[root]# chown root:gtalk {GTALK_HOME}/bin/gtstart
[root]# chmod u=rxs,g=rx,o= {GTALK_HOME}/bin/gtstart
```

Your Gtalk system should now be all set to go. The above commands setup the initial accounts, including the guest and administrative accounts. The password that you've assigned on the 'gtsys' account is the password that is used for the account '000' within Gtalk. This is the initial administrative account. It also sets permissions upon the files so that they are owned by gtalk, except for `gtstart`. By setting `gtstart` to be owned set-uid by root, it allows the user gtalk to start the server.

Part 3. Initial Setup

Starting Gtalk

Gtalk's main daemon process is named `gtalk`. A start-up script named `gtstart` spawns it with any parameters that are provided within. Start Gtalk at this time, without setting any parameters, by running the following:

```
[gtalk]$ {GTALK_HOME}/bin/gtstart
Gtalk Server v1.6.9 Revision 6.
Initializing...
Warning: No port specified.  Using default.
Server: Using port [4000]
Server: daemonizing...
```

The server will background itself once it has loaded, and you will be returned back to the command prompt. To verify that Gtalk is running, you can run the following command:

```
[gtalk]$ ps ax |grep gtalk
   1344 ?        S      0:00 /home/gtalk/bin/gtalk
```

If the output is similar to above, then Gtalk is running.

Logging into Gtalk

Now that Gtalk is loaded and listening on port 4000, as shown in the start-up output, you'll need to use telnet to connect to the server.

```
[gtalk]$ telnet localhost 4000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Welcome to Gtalk: [ ANSI Detected ]
Press <RETURN> for Guest access.

User ID:
```

You should now be at a prompt that is asking for your user ID. Since no other accounts have been created, login under the default account of '000'.

```
User ID: 0
Password: (gtsys password)
```

One nice feature of Gtalk limited online time. For login, you will have sixty seconds to enter your user number and password. If you fail to respond before the online time value has expired, the server will drop your connection with the following message:

```
Your Online Time Has Expired
```

If your connection drops, simply reconnect to try logging in again. If you have entered the correct login and password, your current mail status will be displayed and then you'll be placed into the default channel of main. Also, your login time will be reset and increased to the limit of your class. By default, the SYSOP class has unlimited time online.

```
[     You have no new email      ]
[   You have no new local mail   ]

Your IP is: 127.0.0.1
--> Node [01]: Login (MAIN) at 06:39:46
--> #000:(The Sysop)
```

Try typing 'Hello World!' and hit enter:

```
Hello World!
#01:(The Sysop) Hello World!
```

Congratulations - You just sent a message to the channel! Let's change the handle away from the default of 'The Sysop'. To run a command, first enter the '/' character followed by the (non-case sensitive) command you wish to run. To see who is on, use '/S'. To change your handle, you would want to type '/H *new handle*'. *'new handle'* is an argument that is passed to the command.

```
/h G0D
--> Handle Changed.
I AM G0D!
#01:(G0D) I AM G0D!
```

A full list of commands is available by running '/?'. Extended help information about a command is available through '/? *command*'. As a Sysop, you have access to all commands within the system. This is a short list of commonly used Sysop commands:

/G	Alter a users online time limit
/IP	List the current IP of all users on the system
/K	Kick a user off the system [aka: kill or boot]
/MAKE	Alter a users privilege flags while they are online
/RL	Relog a user to send them back to the login prompt
/SYSCONFIG	Enter the system configuration menu
/U	Enter the user editor

Setting the system passwords

A sysop can reset the system passwords from within the '/SYSCONFIG' command. Set them now so that you don't forget to, as these passwords protect sensitive systems administrative commands. Since the passwords aren't set to anything by default, any password (including just hitting enter) will work for the system password prompts. Run /SYSCONFIG to load the system configuration menu.

```
/SYSCONFIG
Enter current configuration password: (hit enter)

Gtalk System Configuration:

A) System information
B) System toggles
C) System passwords
D) System strings
E) System files (currently uncoded)

Select Option (Q to Quit): Q
```

As with many of the menus inside Gtalk, this menu is *hotkey* sensitive. A *hotkey* menu does not require you to hit enter. Simply press the key and you'll be sent on your way. If you do hit enter, the shown default will be chosen. Press 'c' at the menu prompt to set the passwords. Once the new menu appears, set each one of the passwords listed, one through four. Hit the enter key when asked to enter the current password, since it hasn't been set yet. You will then be asked for the new password twice. This will check for typographical mistakes, as you will need to enter the password exactly the same both time. Once you have set all four, hit 'q' twice to quit the System Configuration menus.

```
Select Option (Q to Quit): q

--> Leaving Gtalk configuration
--> #01:(G0D) joined (MAIN) from GT Config
```

Creating a new class

After setting the system passwords, you'll be placed back into the channel you left from. Now you can continue by creating a new class for your users. Enter the user editor by using '/U' and enter the class editor with the selection of 'C'.

```
/U
Enter current user editor password: (User Editor Password)

User Maintenance (? for Help): C
```

Your new class is going to be designed for normal level users. This will allow you to create accounts for normal users and not need to worry about them doing things they aren't suppose to, such as booting other users off the system. You can call your new class 'USER'. Following along as shown here:

```
Class Edit

CLASS: UNKNOWN
CLASS: GUEST
CLASS: SYSOP
CLASS Name or Number to Edit : USER
ERROR LOADING CLASS
Would you like to create a new one?  Yes/No (Y/N): Y(enter)

What other Class would you like to base it on? (return for none)
--> (enter)
```

The actual editor should now be displayed. You can use Control-N and Control-P to change which field is selected. The default selection is "Class [N]ame". It is a static field and cannot be modified. Use Control-N to move to the field "Time/Properties" and hit the enter key to load the lower menu, such as the example shown here:

```
Class [N]ame   : USER
[P]riority

<Time/Priorities>   <Privs>            <Misc>             <Logs>


Editing Priorities/Time for Class [USER]

[P]riority           : 60
[T]ime               : 15
[A]dded Time         : 15

Priority/Time Edit (P,T,A,Q,?):
```

At this menu, you can alter the class's time online and their priority level. Give this class a priority of 30 (P) and an online time of 60 minutes (T). Set their default time added (A) to be 15 minutes. The time added value is used when a sysop performs a '/G' upon them with no options. Enter 'q' to quit the menu once you've set the three values.

Now that you're back in the top part of the menu, use Control-N to select "Privs". This section defines what access the class has access to. Walk through the menus to familiarize yourself with what commands and access flags are available. Give this class whichever flags you feel are appropriate. Typically, a 'USER' class has the following flags marked as enabled:

```
CMD_H       CMD_T       CMD_P       CMD_Q       CMD_?       CMD_FB
CMD_S       CMD_D       CMD_M       CMD_ANSI    CMD_BBS     CMD_X
CMD_MAIL    CMD_INFO    CMD_PAGE    CMD_INFO    CMD_CLEAR   CMD_PASSWD
CMD_ADDCHANNEL
```

Hit 'q' when finished to return to the main menu display. Next you'll need to set the brackets for which the class has. Use Control-N to select "Misc", hit enter, and the following menu will appear:

```
Editing Staples

#00:(User Handle) This is a sample statement.
--> #00:(User Handle) has arrived.

1 - Starting bracket of channel statement
2 - Ending bracket of channel statement
3 - Starting bracket of action
4 - Ending bracket of action
Q - Quit

Enter Selection:
```

For now, we'll set the users brackets to be '[' and ')' to differentiate them from other classes. Set one and three to '[' and two and four to ')'. Hitting only enter will display the sample and menu again. The sample should now look as follows:

```
#00:[User Handle) This is a sample statement.
--> #00:[User Handle) has arrived.
```

Hit 'q' when finished to return to the main menu display. Use Control-W to begin writing the class. You'll need to select 'S' to confirm that you wish to save the class and quit.

Creating a new user

Now that you have a class setup for your user, create a new user by selecting 'N' at the User Maintenance prompt (where you should now be).

```
User Maintenance (? for Help): N

User Editor New User Menu
Are you sure you want to create a NEW user?  Yes/No (Y/N): Y(enter)

CLASS: UNKNOWN
CLASS: GUEST
CLASS: SYSOP
CLASS: USER
Enter Class Name for new User.
--> USER
Class loaded successfully

Enter User Number to Create : 123
Creating New User [123]
Creating New USER

Enter Name       : User's full name
Enter Street     : User's street address
Enter City       : User's city
Enter State      : User's state
Enter Postal Code: User's zip code

Enter Phone 1    : (000)000-0000
Enter Phone 2    : (000)000-0000

Enter Birthdate  :
MM/DD/YY: 00/00/00

Setting User Enabled Flag
Setting User Number
Setting UNIX login to: [u123]

--> Are you sure you want to save this user? Y(enter)

Cleaning User Directories...
Done.
--> User account saved
--> Creating UNIX login...
--> Setting Password
--> Changing password for UNIX login [u123]

Changing password for user u123.
New password: (enter password)
Retype new password: (enter password again)
passwd: all authentication tokens updated successfully.
Done
```

Once it's completed, you should see 'Done' followed by the User Maintenance prompt.

Congratulations! Now that you know how to create classes and users, you're all set to open the doors for your users! Please be sure to read the contact information in Appendix B in case you have any questions or if you'd like to comment upon this documentation. Also, speaking of doors, continue to read to learn more about setting up online games.

Part 4. Setting Up Features

Dosemu and Doors

Setting up dosemu to allow for playing door games is a bit of tricky business. Make sure you've read all dosemu documentation and HOW-TOs as they'll be very helpful. Hold off on actually building dosemu, as you'll need to apply a patch to assist with security. For this installation example, all of the respective files will be installed into their default locations, set in both `Client/ndoors.h` and `sdos.c`. You will need to modify these files and rebuild them if you wish to change their location. These default locations are:

Sudo-dos (sdos)	/usr/bin/sdos (setuid root)
Dosemu (dos)	/usr/bin/dos
Dosemu (image)	/var/lib/dosemu/image

Our experience with running door games is that you're best running them, unfortunately, as root. Some door games will refuse to run without doing so. To facilitate this, and to work around a number of issues with the dosemu binary being setuid root, a wrapper program was written, `sdos.c` (see note above about hard coded paths). Compile the program, and install it as follows:

```
[user]$ gcc –o sdos sdos.c
[user]$ su - root
Password: (root password)
[root]# chmod u=rxs,g=x,o=x sdos
[root]# cp sdos /usr/bin/sdos
```

Along with `sdos.c` is the file `sudoers`. If you do not have a `sudoers` file, simply use this one. If you do, modify your existing one to add the entries shown in the included example file.

Run the following commands to begin building the dosemu image:

```
[root]# mkdir –p /var/lib/dosemu
[root]# cd /var/lib/dosemu
[root]# tar –zxf {source}/Targets/dosemu.tar.gz
[root]# chown –R gtalk:gtalk .
```

The `dosemu.tar.gz` file contains files that have been written specifically to interact with files written by the Gtalk client. It also includes configuration files and scripts for dosemu that will assist in operating and maintaining the dosemu image. Look at the files that have been extracted and make any necessary path changes if you are not installing in the default locations.

When building dosemu, two versions should be built. One version will need to be *without* our patch applied (`/usr/bin/dos.untouched`) and one *with* it applied (`/usr/bin/dos`). The patch (`/var/lib/dosemu/cntrl.patch`) is *vital* in preventing users from using Control-C to gain access to the dosemu shell. As mentioned before, read all of the dosemu documentation before building it. The following is a line-by-line example of our build:

{TODO}

Now apply the patch and rebuild:

{TODO}

Dosemu should now be all set for installing door games. Read your door game instruction files to see how they should be installed. Use the provided script named `local-dos-conf` to obtain shell access within dosemu. For our example, we'll be installing BRE:

{TODO}

Once a door game has been installed within dosemu, the door menu within Gtalk needs to be configured. The configuration menu allows you to set the door name, location, type, and other aspects shown below. A user with the flag DOOR_MODERATOR will have access to the setup menu within '/DOORS'. Here is how we will setup BRE:

```
/DOORS
...
Enter Choice: S(enter)
...
Choose door number: 0

Door Record #00
1) Name     :
2) Location :
3) Dos cmd  :
4) Lockfile :
5) Multinode: No
6) Priority : 0
7) Options  : None
8) Type     : Door.sys

E) Enabled  : No
Q) Quit

Command: Q
```

When prompted to choose the door number, enter a value between zero and eight. There is a hard coded limit of nine doors. After selecting the door record to modify, the above hotkey menu will be shown. Review the table below for the menu definitions:

Name	This field designates the string that will be shown to the user upon entering the door menu, when the doors are listed.
Location	This field is used to set the users location, as would be shown in the '/S' menu. Gtalk also uses this field when determining if another user is in the door, by comparing this string with the location string of the other users online.
Dos Command	This field sets the command that should be run inside the Dosemu to run the door. Typically, a batch file is specified here that properly loads and executes the door.
Lockfile	This option is used to specify if there is a lockfile that the door uses. If this is set and the lockfile exists, no users will be allowed in the door, if there is a valid user online in the door. Otherwise, the file is removed and the user is allowed to proceed.
Multinode Capable	This option is a toggle (Yes/No) that specifies if the door is capable of allowing multiple people at the same time.
Required Priority	This option specifies the required priority to see and enter the door. If the user does not have a high enough priority, they will not be able to enter.
Options	This is a mini-hotkey menu that will give you the option of setting two different flags. One flag (Moderator) will require that the user have the DOOR_MODERATOR flag. The other will toggle passing the user's node number to the Dos command.
Type	Another mini-hotkey menu to choose if the door is a door.sys or a chain.txt type.
Enabled	This option is a toggle (Yes/No) that specifies if the door is enabled or disabled.

For BRE, an example configuration would be as follows:

```
Door Record #00
1) Name     : |*FFBRE Door Game
2) Location : BRE
3) Dos cmd  : C:\DOORS\BRE\START.BAT
4) Lockfile :
5) Multinode: Yes
6) Priority : 30
7) Options  : (Node Number)
8) Type     : Chain.txt

E) Enabled  : Yes
Q) Quit

Command: Q
```

Make sure that you quit so that it saves the record to the data file.

Appendix

A) Definitions

BBS: BBS stands for Bulletin Board System. Gtalk's BBS is similar to the message bases in earlier BBS systems. The bases are setup such that a user can post a message for others to read and reply to.

Channels: The Gtalk system works on a basis of channels, similar to AOL's chat rooms. Once a user has joined a channel, they can type in a message and hit return, and it is sent to all the users within that channel. Likewise, they can simply sit and watch the messages that other users write to the channel. The default channel that all users first enter is main.

Class: A class is a type of user, such as SYSOP or GUEST. A class record contains information such as time limits, priority level, access flags, and what commands that type of user has access to.

Command: A command is an instruction sent by a user. When a user is in a channel, they can execute a command by typing a '/' followed by the command name. Command names are not case-sensitive, and some may require arguments. A list of commands that a user can access is available by typing '/?'. More information about a command is available by typing '/? *command*'.

Hotkey: A hotkey is a type of menu that does not require you to hit enter after typing your selection. These menus have selections controlled by one key on the keyboard. After pressing that key, that key's function will be executed. Generally, a hotkey menu has a default letter displayed. Hitting enter will execute this default.

Priority: Gtalk maintains access levels on the basis of a priority number, between 0 and 255, stored within their class. The lower the number, the more important they are. A priority of zero is the most powerful and should be reserved for SYSOP.

User: A user contains information regarding the person who is connected. It maintains their online statistics, handle, toggles, real life information, and which class they are associated with.

B) Contact Information

Please feel free to contact the coders of this project by one of the following methods:

- Our mailing list at
  http://groups.yahoo.com/groups/opengtalk
- Our current developer at:
  u550@phringe.org

If you would like to report a bug, please make sure you include the following:

- Your name and email address
- Version number of your Gtalk system
- Screen capture of the bug in action
- Instructions to reproduce the bug
- Core dump files (if any)