ipqc

The IPQ run-time administration graphical user interface

Peter Linich
31st of March, 2000

While administrators familiar with the basic command set of the IPQ program can use utilities such as TELNET to examine accounts and make run-time changes, the ipqc program provides a simple point-and-click interface and can, in the majority of cases, achieve the same result.

![ipqc GUI](image)

**Figure 1 - ipqc GUI**
Before using ipqc users should have a basic understanding of the command-set of IPQ, the command result codes and the consequences of each command. ipqc operates directly on the running account database so any changes made have immediate, and sometimes, irreversible effect.

The GUI consists of five panels. They are:

1. The **identity** panel. This panel contains the controls and fields used to view and make changes to individual accounts,

2. The **update-identity** panel. This panel contains the additional controls used when making adjustments to the traffic counters of an account. The account identity is taken from the **identity** panel,

3. The **alias** panel. This panel is used to create an account alias,

4. The **button** panel. This panel contains miscellaneous buttons, including the **QUIT** button, used to exit ipqc,

5. The **log** panel. This panel contains a text window and scroll bars. Any messages received from IPQ which are not otherwise interpreted by ipqc are displayed here. They include asynchronous update messages, command results, etc.

Many of the functions of ipqc are dependent on the filling-in of different entry fields by the user, and on a connection being open to IPQ. When first started many of the GUI buttons are disabled. The conditions required for each button to be enabled are discussed below.

Each IPQ account record has, at least, an identity, an index and a type. The identity is either an account name (like a user name) or the IP address of the host which it represents. The index is simply the index number of the record in the table of account—the first record is record zero. The record type is one of the following:

<table>
<thead>
<tr>
<th>Record type code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>Host</td>
</tr>
<tr>
<td>N</td>
<td>Account/user-name</td>
</tr>
<tr>
<td>h</td>
<td>Host alias - points to real account</td>
</tr>
<tr>
<td>n</td>
<td>User-name alias - points to real account</td>
</tr>
<tr>
<td>U</td>
<td>Unused/empty record</td>
</tr>
</tbody>
</table>

The identity, index and type of a record are displayed in the top-left corner of the GUI. The identity can be entered by the user, or will be set as a result of a read
operation.

The alias field contains—when the current record is an alias—the index number of the real¹ account to which the alias points.

Underneath the identity field and the other fields and buttons alongside it, there are the record counts and limits. The top row of counts and limits is for incoming traffic. Underneath that is the row for outgoing traffic.

Each counter consists of a READ&RESET button, a count and a limit. Normally the READ&RESET button is disabled.

The count shows the traffic count. If the count exceeds the corresponding limit the background of the count will be shaded red. If the count is under the limit it will be shaded green. The count is in megabytes.

The limit is in megabytes. A value of zero indicates no limit.

**Reading account records**

Any of the four "read" buttons can be used to read and display an account record. They each work in different ways:

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>READ</strong></td>
<td>This reads the record specified in the identity field. Both a connection to the IPQ box must be open and the identity field must be non-blank for this button to be available.</td>
</tr>
<tr>
<td></td>
<td>If the record is an alias the real account to which it points is displayed. This will cause the identity field to change to the identity of the real account.</td>
</tr>
<tr>
<td><strong>READ UNALIASED</strong></td>
<td>This reads the record specified in the identity field. Both a connection to the IPQ box must be open and the identity field must be non-blank for this button to be available.</td>
</tr>
<tr>
<td><strong>READ FIRST</strong></td>
<td>This reads the first account record (record zero). A connection to the IPQ box must be open for this button to be enabled.</td>
</tr>
<tr>
<td></td>
<td>This and <strong>READ NEXT</strong> are the only buttons which can cause an unused record to be displayed.</td>
</tr>
<tr>
<td>Button</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>READ NEXT</td>
<td>This reads the account record following the one currently displayed. The record index is determined by incrementing the displayed index. A connection to the IPQ box must be open and a record must be being displayed for this button to be enabled.</td>
</tr>
</tbody>
</table>

When any of the read buttons is clicked all the fields in the identity panel, except for the identity itself, are cleared. They are filled in with the results of the read operation when IPQ responds.

**Set limits**

The set limits button is used to set all the limits of an account at once. The button is enabled when a connection to IPQ is open, the identity field is non-blank and all the limit fields have valid limit values\(^2\).

When clicked, and as per the read buttons, the entire identity panel, except for the identity field itself, is cleared. Pressing the READ button will redisplay the record.

**Delete**

The delete button is used to delete either a real account or an alias. Aliases can always be deleted. Real accounts can only be deleted when no current connection exists belonging to the account.

The delete button is available when a connection to IPQ is open and the identity field is non-blank.

**Read&reset**

The read&reset buttons are used to issue a read&reset command for a single counter belonging to an account. These buttons are available when a connection to IPQ is open, the identity field is non-blank and the secret key is current.

**Update identity**

The update-identity panel contains the UPDATE IDENTITY button and its associated entry fields. The button is available when a connection to IPQ is open, the incoming and outgoing fields contain valid numbers and the identity field is non-blank.

\(^2\)Valid limit values are zero or positive numbers. Badly-formatted entries will be highlighted in red.

Copyright © 2000 University Of New South Wales, Sydney, Australia
When clicked an update identity command is sent to IPQ containing the supplied incoming and outgoing counts, and containing the optional description field contents. The description field should always be filled-in and describe the reason for the update. The description, and the operation itself, appear in IPQ's logs.

**Alias**

The **ALIAS** button, and its associated fields, are used to create either a host or user-name alias. The real account field and the alias field contain the details of what should be aliased to whom. The button is available when both fields are non-blank and a connection to IPQ is open.

**Log to file and Close log file**

These two buttons are used to, respectively, open a log file to which the responses from IPQ which appear in the log panel are subsequently written, and to close the file.

The **LOG TO FILE** button is always available. The **CLOSE LOG FILE** button is only available when a log file is open.

**Server status and Server help**

These two buttons send, respectively, a status or help command to IPQ. The response is written to the log panel. These buttons are available when a connection to IPQ is open.

**Server shutdown**

This button is only available when a connection to IPQ is open and the secret key is current. This button pops up a confirmation window after which a shutdown command is sent to IPQ. In response IPQ immediately closes all administrative connections (including the one to the running **ipqc**) writes its account database out to disk and exits.

**Secret key**

Some functions of **ipqc** (like the **SERVER SHUTDOWN** button) are protected by a secret key. It's not particularly secret—seeing as by spending time reading the source code you can work out what it is and how to use it—but it does protect against accidental catastrophic use.

Basically, the secret key is made "current" by setting two of the GUI's fields to certain values. It is similar to using a password.
Connection to IPQ

When ipqc is started it attempts to open a connection to IPQ. Hopefully, this will be immediately successful. If not, or if an open connection fails, ipqc will attempt to re-establish the connection until the QUIT button is pressed and the program exits.

During the life of the connection various messages will appear in the log panel. This will typically be messages relating to the connection, asynchronous account notification messages (type 301), or command results.

Type 301 messages are sent to all administrative connections—which is the type of connection ipqc establishes with IPQ—in response to an account going over or back under one of its limits, or when an alias is created. The message contains a full copy of the corresponding account record. When the 301 message is limit-related there will be one for the real account plus one for each associated alias\(^3\).

Note, only authorised hosts may establish an administrative connection to IPQ. Therefore ipqc must be run on an authorised host.

\(^3\)The record sent for an alias contains the identity of the alias but the details (counts and limits) of the real account.