JNIOR Web Access

Dated January 3, 2006
(v2.10.0 and later)

INTEG process group, inc.
11279 Perry Highway, Suite 107
Wexford, PA 15090

www.integpg.com

JNIORsales@integpg.com

PH (724) 933-9350
FAX (724) 933-9333

Revision History

Date	OS Version	Change Description
12/28/2005	v2.10.0	Web Access documented.

Access Control and Basic Authentication

The entire JNIOR website, an area of the website or individual files on the website can be set to require the successful entry of a valid JNIOR username and password before they will be presented. The process is called *Basic Authentication* and it is managed by your Browser. This provides for a very basic level of security. Usernames and passwords are not passed in clear text but are also not securely encrypted. In other words if someone were able to capture your communication packets they would not be able to see your username and password without some effort but the information could still be used to gain access to your JNIOR. The approach is designed to limit the accidental viewing of your login information.

File/Folder Permissions

Files and Folders within the JNIOR file system have a set of permissions associated with them. This provides for restrictions on reading, writing and also the execution of programs. With website access we are only concerned with the *read* permissions as one cannot write files or execute programs.

File and Folder permissions are manipulated in the Command Line mode on the JNIOR available through Telnet access or in the configuration section of the standard applets. In the following examples we will assume that the user has logged into the Telnet session using a valid administrator's username and password.

Viewing Permissions

The **dir** or **ls** commands generate a file listing. The two commands are synonymous and either can be used with the **–l** option to list file and folder details. For instance:

```
TINI /> dir -l
total 9
drwxr-x      1 jnior    admin          6 Nov 09 16:01 .
drwxr-x      1 jnior    admin          6 Nov 09 16:01 ..
-rwxr--      1 jnior    admin       1052 Jan 03 08:40 jniorboot.log
-rwxr--      1 jnior    admin      19797 Jan 03 08:39 jniorsys.log
-rwxr--      1 jnior    admin      25040 Dec 23 22:34 jniorsys.log.bak
drwx---      1 jnior    admin         13 Nov 09 16:01 www
drwxr-x      1 jnior    admin          2 Nov 09 16:01 etc
drwxr-x      1 jnior    admin          0 Nov 09 16:01 tiniext
drwxr-x      1 jnior    admin          0 Nov 16 12:45 flash

TINI />
```

Here we see a representative listing of files and folders at the root of a JNIOR Model 310. This appears in the Unix format for compatibility with FTP clients and will be familiar to some.

The first column displays the permissions. The entries here follow the format ‘drwxrwx’ with each position either displaying the letter or a dash (‘-‘). This can be interpreted as follows:

> The ‘d’ is used to mark folders. In the above example the entries for ‘www’, ‘etc’, ‘tiniext’ and ‘flash’ represent folders which themselves may contain other files and folders. Note that ‘www’ is the default root for the JNIOR website. ‘.’ And ‘..’ are special folders representing the current folder and the parent folder respectively. A dash (‘-‘) in this permission field indicates that the entry is a file.
>
> The first group of ‘rwx’ indicates the read (‘r’), write (‘w’) and execute (‘x’) permissions associated with the *owner* of the file. Generally the owner is the user who was logged in when the file is created. The third column of the listing displays the file’s or folder’s owner. This can be changed using the **chown** command.
>
> The second and last group of ‘rwx’ indicates the read (‘r’), write (‘w’) and execute (‘x’) permissions used when anyone other than the owner accesses the file or folder.

Permission can be manipulated using the **chmod** command. By default the owner is given full access to read, write and execute all his/her files or folders. Therefore the first set of ‘rwx’ typically are always present. By default also, others are allowed to read and execute files or folders but not write them. Therefore the write (‘w’) permission is absent from the second set of ‘rwx’ being replaced by the dash (‘-‘) placeholder.

Note that in the above example listing only the owner (and administrators) have access to the 'www' folder. This is not the default but has been set that way using the **chmod o-r** command. Additional information regarding the use of the **chmod** command maybe be obtained by typing **help chmod**. For instance:

```
TINI /etc> help chmod
chmod [options] FILE

Change the permissions of a file.
Options: [[u|o][+|-][r|w|x]] OR [##]
Examples:
To remove user's read and write permission on s.txt
  chmod -r-w s.txt <OR> chmod 10 s.txt
To remove other's execute and add read permission on s.txt
  chmod o+r-x s.txt <OR> chmod 75 s.txt

TINI /etc>
```

The **help** command supplies syntax information for all JNIOR commands.

Website Access Rules

As mentioned earlier only the read ('r') permission is of interest affecting the availability of website files. In order to access a file using the browser the following must be true:

1) The user must have read access to the file and
2) The user must have read access to the folder containing the file and every parent folder on up to the root.

Therefore by removing the read ('r') permission for others as in the previous example file listing no file can be obtained from the website until the user logs into the site using the username and password of the folder's owner or an administrator.

In this case when your browser is directed to the JNIOR you will be presented with the login screen. Enter a valid username and password and the home page will be displayed. Your credentials (login username and password) are remembered by the browser for this session and used as required allowing you to move about the website.

If the *owner* is an *administrator* then you will have to login using an account with administrator privileges. Otherwise you may login as either the owner or the administrator. Various areas of the website can be protected and set to require different login credentials by setting of folder and file owners and permissions.

For those familiar with Unix file systems you will note that JNIOR does not support user groups.

Authentication and Port Query

The JNIOR Protocol by default requires a successful login. Applets served by the JNIOR website often will then use the JNIOR Protocol to display and control JNIOR I/O. A special CGI script is provided which allows an applet to obtain port and authentication information from the JNIOR as is needed to make the JNIOR Protocol connection. The **query.cgi** script with the **access** parameter is used for this purpose. The response is XML as follows:

```
http://10.0.0.110/query.cgi?access

        <?xml version="1.0" ?>
        <!-- Generated by JNIOR Model 310, INTEG proces group, inc. -->
        <access>
          <authorization>am5pb3I6am5pb3I=</authorization>
          <JniorServer>9200</JniorServer>
          <ModbusServer>502</ModbusServer>
        </access>
```

Where the **authorization** string may be supplied in the LoginRequest over the JNIOR Protocol (see the Protocol Documentation). The **JniorServer** value defines the protocol port for the JNIOR Protocol and the **ModbusServer** value the port for Modbus communications.

If no login has been performed the authorization tag will be absent or null. If the JniorServer has been disabled the JniorServer tag will be absent or null. This is the same for the ModbusServer tag.

Registry Query

The **query.cgi** script can be used to obtain Registry information.

Obtaining a single Registry Key

The **entry** parameter is used to query a single Registry key. The Key is supplied following an equals (‘=’) sign. For instance:

```
http://10.0.0.110/query.cgi?entry=IpConfig/Domain

        <?xml version="1.0" ?>
        <!-- Generated by JNIOR Model 310, INTEG proces group, inc. -->
        <registry query="IpConfig/Domain">
          <entry name="IpConfig/Domain">
            <value>integpg.com</value>
          </entry>
        </registry>
```

If the Registry Key does not exit then a null string is returned:

```
http://10.0.0.110/query.cgi?entry=IpConfig/Unknown

        <?xml version="1.0" ?>
        <!-- Generated by JNIOR Model 310, INTEG proces group, inc. -->
        <registry query="IpConfig/Unknown">
          <entry />
        </registry>
```

Obtaining Matching Keys

The **query.cgi** script may be used to obtain the entire Registry or a selection of matching Registry keys. The **key** parameter is used for this. If any substring of an existing Registry Key matches the parameter given then the Key is supplied in the response.

This example reports all of the Hour Meters:

```
http://10.0.0.110/query.cgi?key=$HourMeter

<?xml version="1.0" ?>
<!--  Generated by JNIOR Model 310, INTEG proces group, inc. -->
<registry query="$HourMeter">
  <entry name="IO/Outputs/rout3/$HourMeter">
    <value>20.24</value>
  </entry>
  <entry name="IO/Outputs/rout5/$HourMeter">
    <value>1.79</value>
  </entry>
  <entry name="IO/Outputs/rout7/$HourMeter">
    <value>1.78</value>
  </entry>
  <entry name="IO/Outputs/rout2/$HourMeter">
    <value>102.88</value>
  </entry>
  <entry name="IO/Outputs/rout4/$HourMeter">
    <value>72.38</value>
  </entry>
  <entry name="IO/Inputs/din6/$HourMeter">
    <value>1.79</value>
  </entry>
  <entry name="IO/Inputs/din8/$HourMeter">
    <value>1.78</value>
  </entry>
  <entry name="IO/Inputs/din1/$HourMeter">
    <value>53.90</value>
  </entry>
  <entry name="IO/Inputs/din3/$HourMeter">
    <value>20.24</value>
  </entry>
  <entry name="IO/Inputs/din5/$HourMeter">
    <value>1.79</value>
  </entry>
  <entry name="IO/Inputs/din7/$HourMeter">
    <value>1.78</value>
  </entry>
  <entry name="IO/Inputs/din2/$HourMeter">
    <value>102.88</value>
  </entry>
  <entry name="IO/Inputs/din4/$HourMeter">
    <value>72.38</value>
  </entry>
  <entry name="IO/Outputs/rout6/$HourMeter">
    <value>1.79</value>
  </entry>
  <entry name="IO/Outputs/rout8/$HourMeter">
    <value>1.78</value>
  </entry>
  <entry name="IO/Outputs/rout1/$HourMeter">
    <value>53.90</value>
  </entry>
</registry>
```

If a blank **key** parameter is supplied then the whole Registry is returned. Refer to the Registry documentation for details on the available keys and their usage.

Note: Special characters in URLs may be %xx encoded. Parameters are not quoted.