Mindoo FTP Server

Version 1.0.0

The Mindoo FTP Server project integrates the Apache FtpServer (http://mina.apache.org/ftpserver-project/) into IBM Notes/Domino R8.5.3 or higher.

The project consists of three parts:
An OSGi plugin on the server side that contains the actual FTP server and all required libraries, a Dojo 1.8 based graphical user interface for web browsers as well as a classic user interface for the Notes Client, both contained in a configuration database.

The configuration database is used to define ports/hostnames for the server, lists to block specific FTP clients or whole IP ranges and lists of users, their access rights (read/write/which directory) and password hashes.

Main Features

• Configurable read and write access for FTP clients
• Users can store files in their personal folder or in folders shared with other users
• Idle timeouts, max concurrent logins and upload/download rates per user
• Resuming of uploads and downloads
• Access can be restricted by defining a time range
• Support for passive FTP mode (e.g. FTP server running in VM on VMware™ server)
• Manual block lists and automatic block list creation after too many failed logins
• Extensive logging of logins and login attempts in log database
• FTP Server is running in OSGi framework of Domino HTTP task 8.5.3 or higher
• Launches on HTTP task startup (using XPages Preloader technology)
• Dojo 1.8 based web UI for Domino R9, classic Notes Client UI available for older servers/clients
• Server deployment via Eclipse Update Site database

Installation

Plugin deployment
The easiest way to deploy the plugin on the Domino server is by importing it into an NSF update site database, as described in this wiki entry: http://www-10.lotus.com/ldd/ddwiki.nsf/dx/XPages_Extension_Library_Deployment

After setting the Notes.ini variable OSGI_HTTP_DYNAMIC_BUNDLES and restarting the HTTP task via „restart task http“, you can verify that the plugin is installed with the console command:

tell http osgi diag com.mindoo.ftp
There should be no error displayed in the server console.

**Configuration and log databases**
Two databases need to be created and signed on the server: the configuration database and the log database.  
By default, the configuration database is expected to be on the top level of the Domino data directory and named

```
    mindoo-ftp-config.nsf
```

For the log database, the default filename is

```
    mindoo-ftp-config-log.nsf
```

By setting Notes.ini variables, you can change these locations:

```
$mndFTPConfig=path/to/ftpconfig.nsf
$mndFTPLog=path/to/ftplogs.nsf
```

or as an alternative, you can also specify a different server:

```
$mndFTPConfig=MyServer/Company!!path/to/ftpconfig.nsf
$mndFTPLog=MyServer/Company!!path/to/ftplogs.nsf
```

**Set folder on local disk for FTP files**
For performance reasons (and to support resuming for uploads and downloads), FTP files are stored on the local server disk.  
The folder to be used needs to be set via the Notes.ini variable $mndFTPBaseDir:

```
$mndFTPBaseDir=c:\mftp
```

If this variable is missing, the server is using a subdirectory “mftp” of the system temp folder as the base path as a fallback and displays warning messages in the console.

**ACL**
Users that need to modify the FTP server configuration need at least editor access to the configuration database.  
Log documents are written with the server ID privileges, so no other special user needs to be listed in the ACL for the FTP server to write logs. Web configuration UI users need at least read access to the log, otherwise the two tabs that display logs won’t work.

**Configure automatic server start**
Add the following lines to the Notes.ini to start the FTP server automatically when the HTTP task starts:

```
XPagesPreload=1
XPagesPreloadDB=mindoo-ftp-config.nsf/ftplauncher.xsp
```

Modify the path to the config database to fit your local setup.
Web configuration user interface

Open the configuration database in a browser (tested with Firefox 21 and Chrome 27):


The web UI contains four tabs: the server configuration, user configuration and two tabs to scroll through log documents:

![Server configuration](image)

Server configuration

Create a new server configuration by clicking “New”:

![FTP Configuration](image)

Please fill out the fields marked with an asterisk:

- Domino Server
• Port
• Max idle in sec (the default value should be fine)

By default, Auto Blocking is enabled. This means that the server counts the number of failed logins for an IP address. If this number exceeds the configured thresholds (e.g. 20 failed logins in 30 minutes), the IP is automatically added to the Auto-blocked addresses list and a login will no longer be possible for this IP until it is removed from the list.

Next, click the “Enabled” checkbox so that the FTP server can see this document when it scans for configurations.

Click “Save and close” to save the configuration and close the dialog.

**Restart FTP server**
The following step is only required after new server configurations are created or port/hostname values have changed.
It is not required when IP block list entries were changed.

There are four server console commands to control the FTP server:

```
tell http osgi mftp status
tell http osgi mftp start
tell http osgi mftp stop
tell http osgi mftp restart
```

What they do is pretty self-explanatory: you can check if the server is running (and which port it uses) and start/stop it.

To make the server use the new settings, enter this command:

```
tell http osgi mftp restart
```

Please note that this command only restarts the FTP server and not the whole HTTP task.
Since the FTP server is running within the HTTP task, when you restart the HTTP task, the FTP server is also restarted.

That’s it, the FTP server is up and running!

**User configuration**
Go back to the web user interface, navigate to the users tab:
and click “New” to create your first user document.

Again, please fill out the fields with an asterisk:

- Directory on server (relative to the configured base directory)
- Login: Username for FTP client login
- Name: Display name for this user, only used in views so far
- Password: contains the hashed FTP login password

Activate the “Enabled” checkbox and click “Save and close” to see the new user document in the grid:

The first column of the grids for server configurations and users displays the enabled state and whether a time range has been defined for a user:
Logging

The tabs „Errors and warnings“ and „All Logs“ display the contents of the log database sorted by date:

You can use the action “Clear log” to remove all log entries for the currently selected server.

Notes configuration user Interface

The user interface for the Notes Client provides almost the same functionality as the web UI:
### FTP Server Configuration

#### Users by name

<table>
<thead>
<tr>
<th>Server</th>
<th>User</th>
<th>Login</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disabled User</td>
<td>disableduser (rw)</td>
<td></td>
<td>Disabled users are marked with a red bullet in the user list.</td>
</tr>
<tr>
<td>Tester</td>
<td>testlogin (rw)</td>
<td></td>
<td>This is a test account to see if the FTP server is working.</td>
</tr>
<tr>
<td>Time Range User 1</td>
<td>timerangeuser1 (rw)</td>
<td></td>
<td>We defined a time range for this user and are within that range.</td>
</tr>
<tr>
<td>Time Range User 2</td>
<td>timerangeuser2 (r)</td>
<td></td>
<td>Another time range user, this time with read only access and range is in the past.</td>
</tr>
</tbody>
</table>

#### FTP Disabled User

**Domain Servers:**
- Server/Mindoo

**Directory on server:**
- folder1

**Limited access:**
- Use time-controlled access

**Max idle time in sec:**
- 200

**User:**
- Login: disableduser
- Name: Disabled User
- Password: [redacted]

**Permissions:**
- Read: yes, Write: no

**Concurrent logins:**
- Max: 5
- Max concurrent logins per IP: 5

**Transfer rate:**
- Max download rate: 0
- Max upload rate: 0

#### Description

Disabled users are marked with a red bullet in the user list.
Tips

Using port 21 on Linux

The following text was taken from the Apache FtpServer website:

Under Linux only programs running as root is allowed to bind and listen to ports with port numbers below 1024. However, running a server, which communicates with untrusted clients as root is not recommended for security reasons. The standard way to solve this problem in servers such as Apache HTTPD is to start the server as root and bind to the privileged port and then use the setuid C function to change the user ID of the current process. In Java there is no equivalent to the setuid C function in the standard API which means that one would have to use a native library to achieve the same, something which FtpServer doesn’t support at the moment.

So, to have FtpServer listen on port 21 but still run it as a normal user one will have to look at other solutions such as using the firewall built into Linux. It turns out that this is really simple. Using the iptables command we can add a rule to the firewall that rewrites all TCP packets coming in on port 21 so that they are effectively forwarded to port 60021:

```
sudo iptables -t nat -A PREROUTING -p tcp -m tcp --dport 21 -j REDIRECT --to-ports 60021
```

We can now configure FtpServer to listen on port 60021 and it will be available on port 21 as well.
Passive Mode

If the Domino Server is not directly reachable by FTP clients, e.g. because it is running in a VMware Server with NAT port forwarding enabled, the following setup should make the FTP server work:

- **Port:** This port must be reachable from the outside, e.g. by defining a NAT forwarding from public 21 to private 2221.
- **Passive Ports:** Ports on the FTP server machine that are reachable via NAT forwarding (e.g. public 2000 → private 2000).
- **Passive address:** IP of the VMWare VM that is running the Domino server of the FTP server.
- **Passive external address:** Public server IP that clients can connect to.