Access Server

Access servers are proxy servers and the platform firewalls at the same time. They perform the following functions:

* Processing of incoming client connections.
* Packing authorization requests and sending them to the trade server.
* Checking activity of client connections protecting the trade server from attacks and overload (antiflood control).
* Saving history data, depth of market and news, and translate them to clients, thus reducing the load to the history server.
* Cashing and providing Live Update to terminals.
* Monitoring the operation of the history and trade servers.

The unlimited number of access servers can exist for each trade server. Terminals are switched between them automatically, depending on the priority settings.

Structure of Directories and Files

The access server is installed to folder "*access\_server*". This folder contains the following executable files:

* **mt5srvupdater64.exe** — the executable file of the live update system of the access server;
* **mt5access64.exe** — the executable file of the access server.

The main directory of the access server contains five folders: *bases*, *config*, *history*, *liveupdate*, *logs*.

The ***bases*** directory contains the news databases, as well as data on the server performance.

| **Files** | **Description** |
| --- | --- |
| **performance\** | Monthly access server performance databases and data indexes, which are displayed on the Monitoring tab. |
| **news.dat** | Database of news sent to clients. |
| **news.idx** | The index file of the news database. |
| **performance.dat** | Data about the access server performance that are displayed on the "Monitor" tab are written to this file. |

The ***config*** directory contains configuration files:

| **Files** | **Description** |
| --- | --- |
| **server.ini** | Individual settings of the access server. |
| **servers.ini** | Settings of the internal network of servers. |
| **symbols.ini** | Configurations of symbols. |
| **time.ini** | Time settings. |

The ***history*** directory contains the base of history data by symbols, which was received from the history server:

| **Files** | **Description** |
| --- | --- |
| **yyyy.hsc** | Minute data for the symbol, divided by years. |
| **yyyymm.tkc** | Tick data for the symbol, divided by months. |
| **tickers.dat** | Data by tickers. |

The ***liveupdate*** directory contains the latest updates of the client, manager and administrator terminals:

| **Files** | **Description** |
| --- | --- |
| **mt5adm.build** | Live update of the administrator terminal. The build number is specified after the point. |
| **mt5clw.build** | Live update of the client terminal. |
| **mt5ckwide.build** | Live Update of MetaEditor. |
| **mt5clwmql.build** | Live Update of the MQL5 compiler. |
| **mt5man.build** | Live update of the manager terminal. |

The ***logs*** directory keeps files of the access server operation journal, as well crash logs:

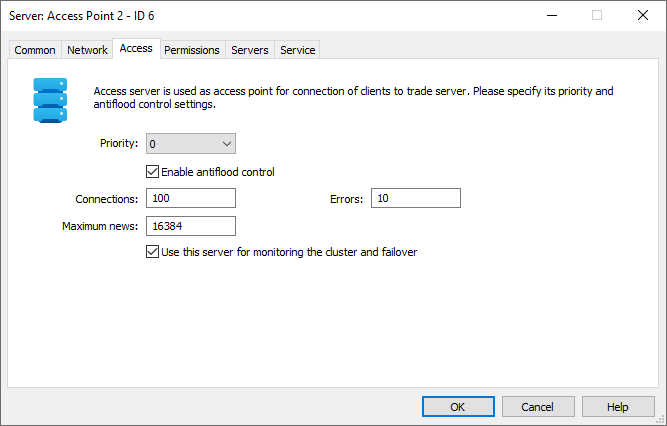
| **Files and folders** | **Description** |
| --- | --- |
| **Crash\crash.log.\*** | The /crash directory contains server crash files. These files are automatically sent to the software developing company for detecting reasons of the crash and eliminating them. |
| **yyyymmdd.log** | Journal files that contain all the information about events that occur on the access server. Server logs are stored in separate files for each working day. Here yyyy — year, mm — month, dd — day. |
| **mt5srvupdater.log** | Journal files of the platform updates. |

# Antiflood Control

The antiflood control system works on access servers. It allows protecting the trading platform from external harmful attacks. This protection system collects the database of users who send incorrect requests to the server (e.g. attempt to authorize with an incorrect login or password), as well as users who send too often requests.

|  |
| --- |
| Antiflood control works with all types of connections to the server including manager ones (via the manager terminal and Manager API). |

The number of connections and incorrect request per time unit can be set up on the "Access" tab of the access server:



## **System of Operation**

The antiflood control system monitors user activity. Users are identified by their IP addresses as well as by the ID linked to their computers and operating systems. The following two activity types are controlled:

* **The total number of connections from one user**  
  The system tracks if a user creates too many connections. When a user connects to the server, their connection counter is incremented by one. If the next connection occurs in less than 3 seconds, the counter is incremented. If the specified time interval is exceeded, the counter is reset. When the counter reaches the number specified in the "Connections" field, the user is blocked for 5 minutes. The blocking period increases if the connections limit is reached again. The maximum blocking period is one hour.
* **Number of invalid packets from one user**  
  The system blocks brute-force attacks by analyzing authentication errors, which imply multiple login attempts with incorrect data. Also, the system detects garbage flood packets, which can be sent to the server by third-party utilities in an effort to reduce the server performance (i.e. a DoS attack). When a user sends an invalid packet to the server, the user's error counter is incremented by one. If the next invalid packet is sent in less than 5 minutes, the counter is incremented. If the specified time interval is exceeded, the counter is reset. When the counter reaches the number specified in the "Errors" field, the user is blocked for 5 minutes. The blocking period increases if the connections limit is reached again. The maximum blocking period is one hour.

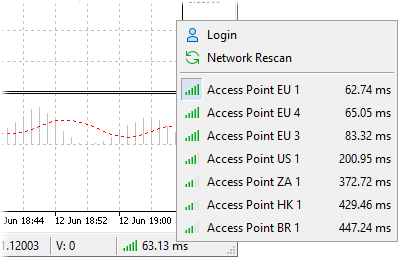
The following records appear in the access server operation Journal when a user is blocked:

* *IP* is blocked after *N* connections [intruder] — a user is blocked by IP if the number of connections is exceeded;
* *CID* is blocked after *N* errors [intruder] — a user is blocked by CID (computer ID) if the number of errors is exceeded.

|  |
| --- |
| * It is strongly recommended to keep the antiflood control system enabled. * IP addresses added to the "always allowed" list in the corresponding section, are not checked by the antiflood control system. |

# Priority

The preference of an access server for client terminals to connect to a trade server is defined by its priority and connection quality. The lower the value if priority is, the more preferable the access server is. A base priority (from 0 to 15) can be specified in its settings. It defines the server preference if all other conditions are equal. The final analysis of an access server is conducted upon the ping and the current priority, which depends on the basic priority and the number of current connections. Also, the quality of connection to the server is shown in the client, manager and administrator terminals based on the same data:



The current priority is calculated according to the following formula: *Current Priority = (Base Priority + Connections / 200)*,

where:

* **Current Priority** is the priority at the server current moment;
* **Base Priority** is the base priority set in its parameters;
* **Connections** — the number of current connections.

Every 200 client connections increase the current priority of a server by one. The value of the current priority of access servers is available on the "Network" tab.