

Accounting Service additional configuration - SSO

Contents

- [Introduction](#)
- [Configuration settings](#)
 - [Recommended changes](#)
 - [Optional changes](#)
- [Notes about editing application configuration files](#)
 - [General notes](#)
 - [About YAML syntax](#)
 - [How to speed up troubleshooting](#)

Introduction



NOTE: The basic configuration for Accounting Service is defined in the SSO setup file `win32.config/unix.config`. See [SSO Installation Accounting Service settings](#) about the properties in the configuration template.

There are couple of settings left to be customised during or after SSO installation for the Accounting Service Spring Boot application in the following file:

```
ubilogin
  custom
    accounting
      config
        application.yaml
```

A file with the default settings is copied from the installation package to this location under `ubilogin-ss0` by the setup script.

Edit the YAML (<https://en.wikipedia.org/wiki/YAML>) file in this location (`ubilogin/custom/accounting/config/application.yaml`) and change default values for properties explained under section [Configuration settings](#) below. Check also some hints in the [notes section](#) below.



After you have made your changes and saved the file execute SSO install / update script which will copy the configuration files to their proper place under `ubilogin-ss0` during runtime before starting the service

```
accounting
  config
    application-install.properties
    application.yaml
  ids-accounting-service.jar
  logs
```

Configuration settings

The settings in the `application.yaml` are organised in a hierarchical way. All Accounting Service specific settings are under `ubisecure.ids.accounting:` section in the file.



Accounting Service has some additional configuration properties for advanced configuration. These can be communicated separately if a need arises for instance with JMS queue handling or event data security algorithm.

In addition, many of the [Spring Boot Common application properties](#) are applicable for the Accounting Service regarding e.g. logging, JMS, or security.

Recommended changes

The following table lists the properties you should always check and change under `ubisecure.ids.accounting:` section.

Property	Description	Default
----------	-------------	---------

calculation-schedule	<p>Defines time and interval when the calculator job will process daily events to calculate daily unique users. Use "cron" format starting from seconds on the left.</p> <p>Silent hours on a daily basis to calculate the previous day's events is recommended.</p> <p>NOTE: In a clustered environment the time to execute should be different on each node in order to avoid unnecessary collisions in the SQL database.</p> <p>Finding the events of a 24 hour day is based on <code>calculation-timespan</code> setting, see table under Optional changes. Always a full 24 hour day is calculated. The first instant of a day is 0:00:00.000 and the last instant of a day is 23:59:59.000 in Accounting Service reporting time-zone which is by default UTC.</p> <p>Monthly calculation happens always on the last day of the month in the same job. Before monthly calculation the system checks that each day in this month has been calculated and if not calculates the missing days.</p>	<p>0 0 3 * * *</p> <p>Each day at 03:00 in the system time-zone</p> <p>Cron format:</p> <p><second> <minute> <hour> <day-of-month> <month> <day-of-week> <year> where <year> field is optional</p>
cleaner-enabled	<p>Defines if the cleaner job to remove obsolete data will be executed as a Boolean value.</p> <p>If <code>true</code>, the cleaner job executes based on the cleaner configuration settings. If <code>false</code>, the cleaner job will never run.</p>	false
cleaner-schedule	<p>Defines time and interval when the cleaner job will be executed. Use "cron" format starting from seconds on the left.</p> <p>Silent hours are recommended.</p> <p>NOTE: In a clustered environment the time to execute should be different on each node in order to avoid unnecessary collisions in the SQL database.</p> <p>See also <code>cleaner-earliest-age</code> setting under Optional changes.</p>	<p>0 0 4 * * *</p> <p>Each day at 04:00 in the system time-zone</p>
csv:accounting-filename	<p>Defines the format for Monthly active unique user counts CSV file name.</p> <p>Supports two arguments: First argument <code>%1\$s</code> is the report month. Second argument <code>%2\$s</code> is the file creation time.</p> <p>Recommended naming is: "<organization>-<installation>-%1\$s.csv" e.g. "MyCompany-B2C-%1\$s.csv"</p>	"MonthlyAccounting-%1\$s-%2\$s.csv"

Optional changes

The following table lists some optional properties you might want to change under `ubisecure.ids.accounting:` section.

Property	Description	Default
audit-log-max-history	Defines the maximum days of history to preserve for the audit log files as an integer value.	366
calculation-timespan	<p>Defines time span between the calculation day in UTC and the day for which the events are included in the daily calculation. Use ISO 8601 duration format without time precision.</p> <p>See also <code>calculation-schedule</code>, which specifies the calculation day, above.</p> <p>About the duration format see e.g.: https://www.digi.com/resources/documentation/digidocs/90001437-13/reference/r_iso_8601_duration_format.htm</p>	<p>P1D</p> <p>Unique users for the previous (full) day</p>
cleaner-earliest-age	<p>Defines the maximum age for events and daily unique users data which are preserved in the database during the cleaning process. Use ISO 8601 duration format without time precision.</p> <p>Minimum value for this setting is 180 days in order to keep data to verify monthly calculations during the billing process. Monthly data is never cleaned by the system.</p> <p>For statistics purposes you may want to preserve the data for a longer period of time or even disable the cleaner. Consider using RDBMS utilities to backup and archive data.</p>	<p>P180D</p> <p>Data of age more than 180 days is cleaned</p>

reporting-timezone	<p>Defines time-zone for specifying the offset of start and end of the day for the events as a time-zone ID supported by Java.</p> <p>This affects to which events are included in the daily calculation or Daily events report. The events are always stored in UTC time-zone in the database.</p> <p>Consider changing this only during the installation process if your system time-zone is not UTC (as recommended).</p> <p>NOTE: You should never change this in the middle of the month when some days have already been calculated for this month.</p> <p>About Java time-zone IDs see e.g.: https://docs.oracle.com/javase/8/docs/api/java/time/Zoneld.html</p>	UTC
csv:daily-users-filename	<p>Defines the format for Daily active unique users CSV file name.</p> <p>Supports two arguments: First argument %1\$s is the report month. Second argument %2\$s is the file creation time.</p>	"MonthlyUsersPerDay-%1\$s-%2\$s.csv"
csv:events-filename	<p>Defines the format for Daily successful authentication events CSV file name.</p> <p>Supports two arguments: First argument %1\$s is the report month. Second argument %2\$s is the file creation time.</p>	"DailyEvents-%1\$s-%2\$s.csv"

The following table lists some optional Spring framework properties you might want to change.

Property	Description	Default
server.use-forward-headers	<p>When running behind a proxy, the caller wants a link to the proxy and not to the physical address of the machine hosting your app. With this setting the absolute links are rendered correctly when the proxy adds conventional X-Forwarded-For and X-Forwarded-Proto headers. If this feature is not needed you may consider setting it back to the value false due security, see https://tools.ietf.org/html/rfc7239#section-8.</p>	true
logging.file	<p>Defines location and prefix of log files.</p> <p>Audit log file has a suffix <code>_audit</code> in its file name.</p> <p>More information about logging in Accounting Service logging.</p>	"logs/ids-accounting"
logging.file.max-history	<p>Defines the maximum days of history to preserve for the application log files as an integer value.</p>	10
logging.level.com.ubisecure	<p>Defines logging level for Accounting Service implementation.</p> <p>Can be one of the following (case insensitive):</p> <ul style="list-style-type: none"> • trace • debug • info • warn • error <p>Note that you can also change the log level when the system is running, see Accounting Service management.</p>	info

Notes about editing application configuration files

General notes

- Never remove the first line of the `application.yaml` which includes the rest of the settings from the `application-install.properties` file
- A setting in `application-install.properties` has a higher precedence than a setting with the same name in `application.yaml`
- Since Accounting Service is a Spring Boot application it is able to read configuration settings in various ways. This is both a benefit and a caveat so pay attention to have the configuration files only in `accounting/config` directory during runtime and not to leave any extra configuration files (old, temp etc.) in the runtime directories

About YAML syntax

- Comments begin with the number sign #
- The following examples are equal:

```
ubisecure.ids.accounting.csv.accounting-filename: "MonthlyAccounting-%1$s-%2$s.csv"
```

```
ubisecure.ids.accounting:
  csv:
    accounting-filename: "MonthlyAccounting-%1$s-%2$s.csv"
```

- Using quotation marks is not always necessary but when using them in a context like above you don't need to think about escaping reserved characters

How to speed up troubleshooting

- If troubleshooting or experimenting you may modify the `application.yaml` file in the runtime location (`ubilogin-ssso/accounting/config`) and restart the Accounting Service in order to put the changes into effect.



When you are satisfied with the contents of `application.yaml` file remember to copy the resulting file into the primary location (under `custom` directory) so that your changes are not lost in further SSO updates.

- It is not recommended but you may try to modify directly the `application-install.properties` file which is generated during the SSO installation and upgrade process based on the `windows/unix.config` settings. Note that most of the settings there needs to be in sync with SSO server



When you are satisfied with the contents of `application-install.properties` file remember to modify the respective settings in the `windows/unix.config` file so that your changes are not lost in further SSO updates.