PANDORAFMS

Installation and Configuration

02-01-2022

Installation and Configuration

Installation and Configuration

This section includes all the requirements to install and configure a Metaconsole and its instances(nodes).

> Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

Installation

Instance and Metaconsole installation must be **hosted in servers connected** both ways.

So, make sure that:

- The Metaconsole can contact the Instances
- The instances can contact the Metaconsole

> Instances do not need to communicate with each other.

To better understand this requirement, take a look at Metaconsole architecture.

The **timezone setting** should be the same. The more synchronized instance and Metaconsole times are, the more accurate the displayed data will be.

For example, if an instance has a 5-minute difference regarding the Metaconsole, when these data are shown in the Metaconsole, the display of the time lapse since the events were generated will be false.

Instances

An instance or node is a common Pandora FMS Enterprise installation made up by its own **server** and **web console**.

To learn more about how to install an instance, visit the following link.

> Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

Metaconsole

A Metaconsole is a Pandora FMS Enterprise installation with a Metaconsole license.

> It is not possible to use the Pandora FMS console and Metaconsole at the same time.

It is necessary to have an active server in order to perform different operations related to the Metaconsole, such as migration, auto-provisioning, service execution, etc.

License Activation

After installing Pandora FMS Enterprise console, regardless of the installation method, access the Pandora FMS console (http://IP/pandora_console/) and a welcome screen will appear to activate the license. To find out more about how the license is activated, visit the following link.

> In order to activate the Metaconsole, you need a Metaconsole license. If you activate the node license, the normal console will appear.

Metalicence

From Pandora FMS version 7.0NG, there is a unique license for an environment with Metaconsole. It is possible to create as many instances as needed, as long as the total number of agents inside the Metaconsole is not exceeded.

This license is applied in the Metaconsole and can be synchronized in as many instances as desired, allowing centralized management of different agents that will be deployed in those instances.

With this license, if an installation is started from scratch, first install the Metaconsole validating its metalicence. Once validated, register each and every one of the desired instances (explained in the following sections), later synchronizing the metalicence so that you may work on all of them.

Apart from occasional network failures, Pandora FMS nodes must be able to reach Pandora FMS Metaconsole at all times. If you need to support nodes that may go offline for arbitrarily long time periods, contact Ártica ST at mailto:info@artica.es.

Configuration

> Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

In order for instances to communicate with the Metaconsole and the other way around, configure both sides correctly.

Metaconsole

Instance registration and configuration

> Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

In the Metasetup section, you may register and configure the Instances to which the Metaconsole will be linked.

To register a new instance, it is required to know a series of parameters referring to the

instance to be handled. If it is the registration of an instance that has not yet been registered through a license, the default data are:

- **Server name**: localhost.localdomain
- **Auth token**: empty
- **Console URL**: http://IP/pandora_console
- **API password**: empty
- **DB host**: database IP
- **DB name**: pandora
- **DB user**: pandora
- **DB password**: pandora
- **DB port**: 3306
- **Control user**: admin
- **Console password**: pandora

Advance Field

To ensure connectivity between node and Metaconsole, configure the connection data manually.

- **Metaconsole DB host**: database IP
- **Metaconsole DB name**: pandora
- **Metaconsole DB user**: pandora
- **Metaconsole DB password**: pandora
- **Metaconsole DB port**: 3306

These fields indicate the configuration of the connection that the *node* will establish against the *Metaconsole*.

If it is a Pandora FMS installation where a valid license has already been included in the instance, obtain these data from the setup of the instance and its database.

Instances can be modified, deactivated and deleted in configured instance view. There are some indicators that check certain information of the setup of each instance. These checks are done when loading this view, but they can also be done individually by clicking on them.

The indicators are the following:

- **Database**: If the instance database has been misconfigured or you do not have the necessary permissions, the indicator will be red and will give information about the problem.
- **API**: This indicator will test the instance API. If it fails, it will give information about the failure.
- **Compatibility**: This indicator checks some requirements between instance and Metaconsole. The name of the instance server, for example, must match its given name in the Metaconsole configuration.
- **Event Replication**: This indicator shows whether the instance has event replication enabled, whether events have already been received from the instance and how long ago the last replication was.
- **Agent cache**: This indicator states that the last node agent and module status have been correctly saved in the Metaconsole database. When there is a change, only said change will be modified in the database.
- **Synchronization**: This indicator refers to the possibility of being able to synchronize the different elements from the Metaconsole to the instances.

The first three indicators must be green in order for the instance to be properly linked and for its data to appear. On the other hand, the Event Replication indicator only gives information about this feature.

An Instance can be well configured, but without replicating its events.

Once you have chosen to replicate the events, all their management will be done from the Metaconsole, keeping instance events as merely informative.

If database encription is enabled, **the nodes and the metaconsole must use the same encryption_passphrase configuration**.

Index Scaling

Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

Most of the synchronization between the Metaconsole and instances is done by name, regardless of the internal ID of the elements, with the exception of groups, tags, alerts, operating systems and module groups, whose IDs must be synchronized.

In order to **ensure that the IDs of the groups, tags, alerts, operating systems and module groups that are synchronized from the Metaconsole do not exist in the instances**, increase the AUTO_INCREMENT value of the tgrupo, ttag, talert_templates, talert_actions, talert_commands, tconfig_os and tmodule_group tables noticeably. That way, you will provide a leeway in the event that elements are created in the instances for reasons beyond the Metaconsole.

Thus, execute in the Metaconsole's database the following query:

```
ALTER TABLE tgrupo AUTO_INCREMENT = 3000;
ALTER TABLE ttag AUTO_INCREMENT = 3000;
ALTER TABLE talert_templates AUTO_INCREMENT = 3000;
ALTER TABLE talert_actions AUTO_INCREMENT = 3000;
ALTER TABLE talert_commands AUTO_INCREMENT = 3000;
ALTER TABLE tconfig_os AUTO_INCREMENT = 3000;
ALTER TABLE tmodule_group AUTO_INCREMENT = 3000;
```

If you suspect that the number of elements of an instance created outside the Metaconsole may exceed 3000, a higher value can be set.

To improve Metaconsole event performance in large environments, it is recommended to add the following indexes to the database:

ALTER TABLE tmetaconsole_agent_secondary_group **ADD INDEX** id_tagente (id_tagente);

ALTER TABLE tmetaconsole_event **ADD INDEX** server_id (server_id);

Report scheduler

Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

It is necessary to install the server packages (Open and Enterprise) in the system where the Metaconsole is installed in order to launch the database maintenance script (**pandora_db**). Make sure that it is correctly programmed for execution in cron every hour (as detailed in the following link.).

If you are going to use on-demand reports (sent by email), program the cron extension execution as it is done in a normal Enterprise console. Usually this is done by typing in the following line into the cron, adjusting the corresponding local paths:

- /5 * * * * <user> wget -q -O - http://x.x.x.x/pandora_console/enterprise/extensions/cron/cron.php » /var/www/pandora_console/log/console.log

For versions prior to 747 the path will be **/var/www/pandora_console/padora_console.log**.

Finally, to configure SMTP to send emails, edit the corresponding parameters in the mail

configuration section, which have the following values by default:

Instances

In instances, there are a series of parameters to ensure the access of your data with the Metaconsole.

Giving access to the Metaconsole

> Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

The Metaconsole accesses an instance in two different ways:

- Remote access to the **Database** to see and edit the data stored in the instances.
- Access to the to the **API** for some actions such as configuration file edition or NetFlow monitoring

The instance should be configured to **guarantee both accesses** to the Metaconsole.

Database

As we have said before, you must know the database *credentials* to configure the instance in the Metaconsole (host, database, user and password).

Besides this, another important point is **grantig permissions** to the user to remotely access the database. It is done with MySQL's GRANT command:

```
GRANT ALL PRIVILEGES on <MetaconsoleDatabaseName>.* to
<UserName>@<HostAddress> IDENTIFIED BY <UserPass>;
```

For example:

```
GRANT ALL PRIVILEGES on PandoraMetaBase.* to adminmeta@localhost
```

```
IDENTIFIED BY pandora;
```

API

The access to the API instance will be guaranteed with the following parameters:

- **User and password**: It is necessary to know a valid user and password in the instance.
- **API password**: It is necessary to know the access password to the API that is configured in the instance.
- **IP list with access to the API**: In the Instance configuration, there is an IP list that could have access to the API. It is possible to use '*' as wildcard to give access to all IPs or to one subnet

Auto-authentication

Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

In some parts of the Metaconsole, there are accesses to the instance Web Console. For example, in the event viewer when clicking on the agent associated to an event (if there is one), it will take you to the view of that agent in the console of the instance it belongs to. Auto-authentication is used for this access. This authentication is done with a hash for which you need a string configured in the instance: *The auto-identification password*. The password is entered in the "Auth token" field of the instance configuration in the Metaconsole.

This configuration is not necessary to configure the instance in the Metaconsole, but without it, when clicking on one of the links that lead to the instance,

you must pass the authentication.

Event replication

Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

In order for instance events to be seen in the Metaconsole, these should have access to the Metaconsole database.

The instances will be replicated from time to time, their events saving the date and time of the last replication to continue from there the next time.

Besides event replication, they will make Metaconsole autovalidation effective. This is for events that are associated to a module, when they replicate the event to the Metaconsole, they will validate all the previous events that were assigned to the same module.

To configure event replication, activate in the Instance Enterprise Configuration section *Event Replication*.

This will be configured:

- **Interval**: How often (seconds) the server will replicate the events generated from the last replication to the Metaconsole database.

   If it is 60 seconds, the first replication will happen 60 seconds after the server was started.

- **Replication Mode**: Whether all events will be replicated or only the ones that are validated.
- **Show list of events in the local console (only reading)**: When event replication is activated, event management is done in the Metaconsole and there is not access to them in the instance. With this option you will have access to a read-only event view.
- **Metaconsole database credentials** : Host, database, user, password and port (if the port is not indicated, the port by default will be used).

> Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

Event replication is done by the server. In the configuration file there should be the following enabled token:

```
# If set to 1, the event replicate process is enabled. 0 by default. (PANDORA FMS ENTERPRISE ONLY)
# WARNING: This process doesn't do anything if is not properly configured from the console setup

event_replication 1
```

> Restart the server to apply any event replication configuration changes.

> If you add in a Metaconsole a new node which already contains lots of events, it could take a lot of time to copy all the events to the Metaconsole

If you want to modify the date from which the node will synchronize events with the target Metaconsole (for example, to force event replication from the current date), execute this SQL query against the node database for Pandora FMS versions older than 5.1SP3:

```
UPDATE tconfig SET `value` = UNIX_TIMESTAMP() WHERE `token` =
```

```
"replication_copy_last_utimestamp"
```

For versions newer than 5.1SP3, execute the following query:

```
UPDATE tconfig SET `value` = (SELECT MAX(id_evento) FROM tevento)
WHERE `token` = "replication_copy_last_id";
```

> ⚠️ If you have activated the child nodes SELinux security, event replication may not work. Disable it.

Autoprovisioning from the Metaconsole

> ℹ️ Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

From Pandora version 7 onwards, you may find in the Metaconsole configuration setup, the option to register the node in the Metaconsole.

You can also check whether it arrives via api to the Metaconsole and whether the node is registered in the Metaconsole.

Indicate the correct credentials for connecting to the node, database and API.

> 💡 The first time this configuration is carried out, it is possible to enter an API password. In case of update, the node password is established.

Advanced options configuration is the one sent to the node to connect to the database.

Metaconsole Additional Configuration

> Version NG 755 or earlier: you will need to configure the use of the Command Center, where all the relevant information is available.

If the node event replication has been activated, the Metaconsole stores event data in its own database. For their maintenance these data can be deleted and/or moved to the Metaconsole history event database. This is done, as in a Pandora FMS instance, through the execution of the database maintenance script that is at*/usr/share/pandora_server/util/pandora_db.pl*. Usually, to launch it the server file is used. But since it is a Metaconsole there is no need for a server. To do this, copy fhe file /etc/pandora/pandora_server.conf from one of the nodes, edit it, and modify the data related to the database (hostname, DDBB name, user and password) and save the file, for example as:

```
/etc/pandora/pandora_meta.conf
```

Create an script at /etc/cron.daily/pandora_meta_db with the following content:

```
/usr/share/pandora_server/util/pandora_db.pl
/etc/pandora/pandora_meta.conf
```

And modify the its permissions through chmod:

```
chmod 755 /etc/cron.daily/pandora_meta_db
```

In order to execute it, it is necessary for you to have installed the necessary packages to execute (even if it does not) the Pandora FMS server and its Enterprise section.

Execute it manually to check that it works and it does not report errors:

```
/etc/cron.daily/pandora_meta_db
```

Metalicence Synchronization

Next, there is an example of how to synchronize the Metalicence between a Metaconsole and an Instance.

First, there is an instance with its own generated and correctly validated key.

Once the node is generated and properly validated, it is configurd in the Metaconsole to later synchronize the license:

Once these steps are finished, the Metaconsole license is "Validated" to synchronize the Metalicence with the Instance.

The result will be the Metalicence in the Instance.

Go back to Pandora FMS documentation index

From:
https://pandorafms.com/manual/ - **Pandora FMS Documentation**

Permanent link:
https://pandorafms.com/manual/en/documentation/06_metaconsole/03_installation

Last update: **2021/09/23 12:22**