



.Disco 開発



m:
<https://pandorafms.com/manual/!776/>
Permanent link:
https://pandorafms.com/manual/!776/ja/documentation/pandorafms/technical_reference/12_disco_development
24/06/10 14:34



.Disco 開発

パッケージ ".disco"

自動検出 を使用すると、公式 Pandora FMS プラグインとカスタムプラグインの両方をロードできます。

カスタムプラグインをロードするには、コンソールと Pandora FMS サーバの両方で次のことができるように、必要なものすべてを含む .disco パッケージを生成する必要があります。

- 新しい検出タスクの設定インターフェイスを表示します。
- 環境内で設定された検出タスクを実行します。

.disco パッケージは、.disco という拡張子を持つ zip ファイルであり、discovery_definition.ini という名前のファイルが少なくとも 1 つ含まれています。

オプションで、.disco パッケージには Pandora FMS コンソールのプラグインロゴとなる logo.png というファイルが含まれる場合があります。ロゴファイルが追加されない場合、コンソールは自動的にデフォルトのロゴを使用します。

最後に、通常、.disco ファイルには、サーバとコンソールに必要なすべてのスクリプト、実行可能ファイル、ライブラリが含まれます。これは必須ではありませんが、プラグインを使用するタスクを構成および実行するシステムで追加のインストール要件を回避するために、一般的であり推奨されます。

したがって、簡単に言うと、.disco ファイルには次の内容が含まれます。

- discovery_definition.ini
- logo.png (オプション)
- スクリプト、実行ファイルおよび、ライブラリ

"discovery_definition.ini" ファイル

discovery_setting.ini ファイルは、プラグイン定義全体が含まれているため、.disco ファイル内で最も重要です。

これには、タスク定義フォームに入力するためにコンソールに表示されるパラメータと、Pandora FMS サーバによるプラグインタスクの実行のための情報の両方が含まれています。

コンソールでの定義と処理を容易にするために、discovery_setting.ini ファイルで使用される形式は PHP の INI 形式です。

discovery_setting.ini ファイルは基本的に 3 つのブロックで構成されています。

- `discovery_extension_definition`
- `config_steps` (オプション)
- `tempfile_confs` (オプション)

共通のコンセプト

以降の説明では、一般的にいくつかの用語が使用されます。この章ではこれらの用語について説明します。

マクロは、情報(値、ファイルへのパスなど)を保存するために使用される一意のテキストキーです。マクロが使用されるとき(実行中など)、マクロに含まれる情報を使用したいことがあります。

有効なマクロは、始まりと終わりがアンダースコア `_` で、間に文字 (`A-Z` および `a-z`) と数字 (`0-9`) のみを含めることができるテキストキーです。

`STRING` は、英数字のテキスト文字列を指します。

`NUMBER` は、数字のみを指します。

`BOOL` は、値 `true / false` / `1 / 0` / `yes / no` を指します。

`VALUE` は、任意の値 (`STRING` / `NUMBER`、または `BOOL`) を指します。通常、同じ型の値を表すためにカンマ区切りのリストで使用されます。

`N` または `M` という文字が大文字で使用されている場合、正の整数を指します。通常、`VALUE` などの他の要素と組み合わせて使用され、複数回繰り返すことができることを示します (`VALUE_N` など)。

サーバマクロ

サーバの実行設定の多くは、マクロの使用に基づいています。主にこれらはユーザによって定義されますが、Pandora FMS サーバに特有の便利なものもいくつかあります。

- `__taskMD5__` タスク ID とアプリケーションの短縮名から生成された、タスクの一意の MD5。これは、実行中に固有の要素(ファイルなど)を生成する場合に役立ちます。
- `__taskInterval__` タスクの実行間隔(秒)。
- `__taskGroup__` タスクグループ(テキスト)。
- `__taskGroupID__` タスクグループ(ID)。
- `__temp__` → `pandora_server.conf` (`temporary`) で設定されたサーバの一時ディレクトリ。
- `__incomingDir__` → `pandora_server.conf` (`incomingdir`) で設定された Pandora サーバの入力ディレクトリ。
- `__consoleAPIURL__` → `pandora_server.conf` (`console_api_url`) で設定された API URL。
- `__consoleAPIPass__` → `pandora_server.conf` (`console_api_pass`) で設定された API パスワード。
- `__consoleUser__` → `pandora_server.conf` (`console_user`) で設定されたコンソールユーザ。
- `__consolePass__` → `pandora_server.conf` (`console_pass`) で設定されたコンソールパスワード。

これらのマクロは、Pandora FMS コンソールでは使用されません。

discovery_extension_definition ブロック

このブロックはプラグインの主な設定を定義し、次のパラメータがあります。

- short_name:
 - 必須。
 - プラグインの短い名前を定義します。
 - 短い名前はユニークでなければいけません。
 - “pandorafms.” で始まる短い名前は、公式 Pandora FMS プラグインによって使用されます。
 - 短い名前には含めることができるのは、アルファベット (A-Z および a-z)、数字 (0-9)、ピリオド (.)、ハイフン (-)、およびアンダースコア (_) です。
- section:
 - 必須。
 - プラグインが表示されるコンソールのセクションを定義します。
 - app, cloud または custom を指定できます。
- name:
 - 必須。
 - コンソールに表示されるプラグインの名前を定義します。
- version:
 - 必須。
 - コンソールに表示されるプラグインのバージョンを定義します。
 - プラグインに変更が加えられた場合は、どんなに小さな変更であっても、バージョンを変更することをお勧めします。
- description:
 - オプション。
 - コンソールに表示されるプラグインの説明を定義します。
- execution_file:
 - オプション。
 - プラグインによって使用されるスクリプトおよび実行可能ファイルへの相対パス (.disco ファイル内) を示します。
 - 指定されたすべてのファイルに実行権限が与えられます。
 - このパラメータは 配列で、そのキーはプラグインファイルの使用を示すために使用される 有効なマクロ になります。
 - このパラメータは 配列であるため、異なるキーを使用して複数回指定できます。
 - 例:

```
execution_file[_exec1_] = "script1.py"
execution_file[_exec2_] = "other/script2.py"
```

- exec:
 - 必須。
 - Pandora FMS サーバがプラグインタスクに対して実行する必要がある実行形式を定義します。
 - このパラメータは 配列 で、そのキーは位置指定になります。つまり、キーは数字で示されることも、示されないこともあります。
 - このパラメータは 配列であるため、異なるキーを使用して複数回指定できます。
 - 定義された実行では 使用されるスクリプトまたは実行可能ファイルへのパスを示したり、そのスクリプトまたは実行可能ファイルのパラメータを示す マクロ を使用するのが一般的です。

- Pandora FMS サーバは、定義された各実行を順番に起動し、すべての実行の結果を保存して、タスク内のプラグインの出力を生成し、そのステータスを取得します。
- 例:

```
exec[] = "'_exec1_' -p '_param1_'"
exec[] = "'_exec2_' -p '_param2_'"
```

- `passencrypt_script`:
 - オプション。
 - コンソールからパスワードタイプのフィールドを暗号化するために使用されるスクリプトへの相対パス (.disco ファイル内) を定義します。
 - 指定されたファイルに実行権限が与えられます。
- `passencrypt_exec`:
 - オプション。
 - コンソールからパスワードを暗号化するためのスクリプトの実行形式を定義します。
 - 上記の実行で期待される結果は、暗号化されたパスワードに対応するテキストになります。
 - 定義内では `_passencrypt_script_` マクロと `_password_` マクロの使用のみがサポートされています。
 - `_passencrypt_script_` マクロは、`passencrypt_script` で定義されたファイルへのパスに置き換えられます。
 - `_password_` マクロは、いつでも暗号化する必要があるパスワードタイプフィールドの値に置き換えられます。
 - 例:

```
passencrypt_exec = "'_passencrypt_script_' --encrypt '_password_'"
```

- `passdecrypt_script`:
 - オプション。
 - コンソールからパスワードタイプのフィールドを復号化するために使用されるスクリプトへの相対パス (.disco ファイル内) を定義します。
 - 指定されたファイルに実行権限が与えられます。
- `passdecrypt_exec`:
 - オプション。
 - コンソールからパスワードを復号化するためのスクリプトの実行形式を定義します。
 - 上記の実行で期待される結果は、復号化されたパスワードに対応するテキストになります。
 - 定義内では `_passdecrypt_script_` マクロと `_password_` マクロの使用のみがサポートされています。
 - `_passdecrypt_script_` マクロは、`passdecrypt_script` で定義されたファイルへのパスに置き換えられます。
 - `_password_` マクロは、いつでも復号化する必要があるパスワードタイプフィールドの値に置き換えられます。
 - 例:

```
passencrypt_exec = "'_passdecrypt_script_' --decrypt '_password_'"
```

- `default_value`:
 - オプション。
 - 実行中に使用されるさまざまなマクロのデフォルト値、つまり、このプラグイン用に作成された各*自動検出*タスクに保存されるデータのデフォルト値を定義します。
 - このパラメータは配列で、そのキーは有効なマクロになります。
 - このパラメータは配列であるため、異なるキーを使用して複数回指定できます。
 - コンソールフォーム (以下を参照) で使用されるフィールドのタイプに応じて、デフォルト値は次のようになります。
 1. string: *STRING* .

2. number: *NUMBER* .
 3. password: *STRING* .
 4. textarea: *STRING* .
 5. checkbox: *BOOL* .
 6. select: *VALUE_N* .
 7. multiselect: [*VALUE_1,VALUE_N*] .
 8. tree: [*VALUE_1,VALUE_N*] .
- タスク設定フォームに含まれる可能性のあるマクロごとにdefault_valueを含めることをお勧めします。
 - 例:

```
default_value[_param1_] = "get_main_info"
default_value[_param2_] = ""
default_value[_param3_] = true
default_value[_param4_] = 0
default_value[_param5_] = "[A,B,C]"
default_value[_param6_] = "[]"
```

次の例は、各ケースのコメントで説明されているすべてのパラメーターが含まれているため、このブロックのベースとして使用できます。

```
[discovery_extension_definition]

; Mandatory
; Defines discovery application short name
; Short name must be unique
; Short names with "pandorafms." Prefixes are used by Pandora FMS for official
applications

short_name = DISCOVERY_APPLICATION_UNIQUE_NAME

; Mandatory
; Defines the section where application will be shown in console
; Possible values:
; apps
; clouds
; custom

section = app

; Mandatory
; Defines discovery application name, shown in console

name = DISCOVERY_APPLICATION_NAME

; Mandatory
; Defines discovery application version, shown in console

version = VERSION

; Optional
; Defines discovery application description, shown in console
```

```
description = DESCRIPTION
```

```
; Optional  
; Defines execution files inside .disk  
; Several execution files can be defined
```

```
execution_file[_EXEC_MACRO_N_] = SCRIPT.pl
```

```
; Mandatory  
; Defines execution for discovery server  
; Several executions can be defined  
; At least 1 is required for server execution
```

```
exec[] = SERVER_EXECUTION
```

```
; Optional  
; Define password encrypt script
```

```
passencrypt_script = PASS_ENCRYPT_SCRIPT.pl
```

```
; Optional  
; Defines password encrypt script execution format  
; _passencrypt_script_ is replaced with the passencrypt_script file path  
; _password_ is replaced with the string (password) to encrypt
```

```
passencrypt_exec = EXECUTION
```

```
; Optional  
; Define password decrypt script
```

```
pasdecrypt_script = PASS_DECRYPT_SCRIPT.pl
```

```
; Optional  
; Defines password encrypt script execution format  
; _pasdecrypt_script_ is replaced with the pasdecrypt_script file path  
; _password_ is replaced with the string (password) to encrypt
```

```
pasdecrypt_exec = EXECUTION
```

```
; Optional  
; Defines the default values for the fields when a new task is created  
; By default all values are empty or not selected  
; Several default values can be defined  
; Possible values depending on field type:  
; string - STRING  
; number - NUMBER  
; password-STRING  
; textarea-STRING  
; checkbox-BOOL  
; select-VALUE_N  
; multiselect - [VALUE_1,VALUE_N]  
; tree - [VALUE_1,VALUE_N]
```

```
default_value[_FIELD_MACRO_N_] = DEFAULT_VALUE
```

config_steps ブロック

このブロックはプラグインタスクの設定手順を定義し、次のパラメータがあります。

これらのルールは、このブロックのすべてのパラメーターに適用されます。

- パラメータは配列であり、そのキーは位置指定になります。つまり、数値で示すことも、示さないこともできます。キーを示すことをお勧めします。
- パラメータは配列であるため、異なるキーを使用して複数回指定できます。
- キーを共有するすべてのパラメータは、同じ要素を参照します。つまり、(パラメータに応じて)異なる値を同じ要素に割り当てます。
- name:
 - 必須。
 - タスク内のさまざまな設定ステップの名前を定義します。
- script_data_fields:
 - custom_fields が同じキーに対して定義されていない場合は必須です。
 - custom_fields が同じキーに対して定義されている場合はオプションです。
 - 動的に生成されるフォームに対して Pandora FMS コンソールが実行する必要がある実行の形式を定義します。たとえば、仮想化環境を監視する予定で、監視する仮想マシンを選択するために仮想マシンのリストを取得したい場合です。
 - 使用されるキーは、このブロックの name 配列に存在する必要があります。
 - 使用されるスクリプトまたは実行可能ファイルへのパスを示すためと、そのスクリプトまたは実行可能ファイルのパラメータを示すために、定義された実行でマクロを使用するのが一般的です。
 - ここで示される実行結果は、次の形式の JSON である必要があります。JSON キーは、CUSTOM_FIELDS ブロックのパラメーターに対応します (以下を参照)。

```
[
  {
    "macro": "_FIELD_MACRO_N_",
    "mandatory_field": "BOOL",
    "name": "FIELD_NAME",
    "tip": "FIELD_TIP",
    "type": "FIELD_TYPE",
    "placeholder": "PLACEHOLDER",
    "show_on_true": "_FIELD_MACRO_N_",
    "encrypt_on_true": "_FIELD_MACRO_N_",
    "select_data": {
      "VALUE_1": "TEXT_1",
      "VALUE_N": "TEXT_N"
    },
    "tree_data": [
      {
```

```

        "name": "TEXT_1",
        "selectable": "BOOL_1",
        "macro": "_FIELD_MACRO_N_",
        "value": "VALUE_1",
        "children": [
            {
                "name": "TEXT_1_1",
                "selectable": "BOOL_1_1",
                "macro": "_FIELD_MACRO_N_",
                "value": "VALUE_1_1",
                "children": []
            },
            {
                "name": "TEXT_1_N",
                "selectable": "BOOL_1_N",
                "macro": "_FIELD_MACRO_N_",
                "value": "VALUE_1_N",
                "children": []
            }
        ]
    },
    {
        "name": "TEXT_N",
        "selectable": "BOOL_N",
        "macro": "_FIELD_MACRO_N_",
        "value": "VALUE_N",
        "children": []
    }
]
}
]

```

- custom_fields:
 - script_data_fields が同じキーに対して定義されていない場合は必須です。
 - script_data_fields が同じキーに対して定義されている場合はオプションです。
 - Pandora FMS コンソールでタスクを作成するときにフォームフィールドから取得されるのと同じ INI ファイルの設定ブロックを定義します。
 - 使用されるキーは、このブロックの name 配列に存在する必要があります。
- fields_columns:
 - オプション。
 - 各構成ステップでタスクフォームのフィールドが分散される列の数を定義します。
 - 使用されるキーは、このブロックの name 配列に存在する必要があります。
 - 値 1 または 2 を割り当てることができます。
 - デフォルト値は 2 です。

たとえば、これは複数の設定ステップを定義する方法です。

```
[config_steps]
```

```
name[1] = First step
script_data_fields[1] = '_exec2_' -p '_param1_' --get_fields"
```

```
name[2] = Mid step
custom_fields[2] = custom_fields_1

name[3] = Last step
script_data_fields[3] = "'_exec2_' -p '_param2_' --get_fields"
custom_fields[3] = custom_fields_2
```

同じ設定ステップに対して `script_data_fields` と `custom_fields` が指定されている場合、フォームには最初に `script_data_fields` によって取得されたフィールドが表示され、次に `custom_fields` で定義されたフィールドが表示されます。

次の例は、各ケースのコメントで説明されているすべてのパラメーターが含まれているため、このブロックのベースとして使用できます。

```
[config_steps]

; Following parameters can be setup for each configuration step
; Several steps can be defined using a different key (N)

; Mandatory
; Defines configuration step name

name[N] = STEP_NAME

; Mandatory if not custom_fields defined
; Defines configuration fields retrieved to console by a command execution
; execution_file scripts can be used to retrieve data
; Command execution output must be JSON as follows
; [
; {
; "macro": "_FIELD_MACRO_N_",
; "mandatory_field": "BOOL",
; "name": "FIELD_NAME",
; "tip": "FIELD_TIP",
; "type": "FIELD_TYPE",
; "placeholder": "PLACEHOLDER",
; "show_on_true": "_FIELD_MACRO_N_",
; "encrypt_on_true": "_FIELD_MACRO_N_",
; "select_data": {
; "VALUE_1": "TEXT_1",
; "VALUE_N": "TEXT_N"
; },
; "tree_data": [
; {
; "name": "TEXT_1",
; "selectable": "BOOL_1",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_1",
; "children": [
; {
; "name": "TEXT_1_1",
```

```
;"selectable": "BOOL_1_1",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_1_1",
; "children": []
; },
; {
; "name": "TEXT_1_N",
; "selectable": "BOOL_1_N",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_1_N",
; "children": []
; }
; ]
; },
; {
; "name": "TEXT_N",
; "selectable": "BOOL_N",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_N",
; "children": []
; }
; ]
; }
; ]

script_data_fields[N] = CONSOLE_EXECUTION_FIELDS

; Mandatory if not script_data_fields defined
; Defines custom configuration fields

custom_fields[N] = CUSTOM_FIELDS_N

; Optional
; Defines the number of fields columns for the configuration steps (1 or 2)

fields_columns[N] = M
```

tempfile_confs ブロック

このブロックは、プラグインによって使用される一時設定ファイルの形式を定義し、次のパラメータがあります。

- file:
 - 必須。
 - Pandora FMS サーバおよびコンソールの実行中に使用できる一時設定ファイルの内容を定義します。
 - このパラメータは 配列で、そのキーはプラグインでの一時ファイルの使用を示すために使用される有効なマクロになります。
 - このパラメータは 配列であるため、異なるキーを使用して複数回指定できます。
 - 実行中にこの 配列 のキーの 1 つが指定されると、この 配列 で指定された内容を含む一時ファイ

ルが生成され、マクロの値はそのファイルの場所に置き換えられます。実行が完了すると、ファイルは削除されます。

- 一時設定ファイル内で、他のマクロを指定して、対応する値に置き換えることができます。
- 例:

```
file[_tempConf_] = "server _param1_
user _param2_
password _param3_
log __temp__/_taskMD5__.log"
```

次の例は、各ケースのコメントで説明されているすべてのパラメーターが含まれているため、このブロックのベースとして使用できます。

```
[tempfile_confs]

; Mandatory
; Defines the content for the temporary file
; File will be used where temporary file macro is specified during executions
; File content replaces fields macros with their values

file[_TEMP_FILE_MACRO_N_] = _FIELD_MACRO_N_,_FIELD_MACRO_M_
```

CUSTOM_FIELDS ブロック

このタイプのブロックには、config_steps ブロック内の custom_fields パラメーターの値として割り当てられている限り、任意の名前を付けることができます。これらはプラグイン設定手順でフォームのフィールドを定義し、次のパラメータがあります。

These rules apply for all parameters in this block:

- パラメータは配列であり、そのキーは位置指定になります。つまり、数値で示すことも、示さないこともできます。キーを示すことをお勧めします。
- 配列であるため、パラメータは異なるキーを使用して複数回指定できます。
- キーを共有するすべてのパラメータは、同じ要素を参照します。つまり、(パラメータに応じて)異なる値を同じ要素に割り当てます。
- macro:
 - 必須。
 - タスク設定値が保存されるマクロを定義します。
- mandatory_field:
 - オプション。
 - フォームの設定フィールドに必須の値が必要かどうかを BOOL 型の値を使用して定義します。
 - 使用されるキーは、このブロックのマクロ配列に存在する必要があります。
 - デフォルトでは、すべてのフィールドが必須です。
 - multiselect および tree タイプのフィールドは必須ではありませんが、要素を選択できないことを認めています。
- name:
 - 必須。

- フォーム設定フィールドの名前を定義します。
- 使用されるキーは、このブロックのマクロ配列に存在する必要があります。
- tip:
 - オプション。
 - フォーム設定フィールドのヘルプを定義します。
 - 使用されるキーは、このブロックのマクロ配列に存在する必要があります。
- type:
 - 必須。
 - フォーム設定フィールドのタイプを定義します。
 - 可能な値は次のとおりです。
 - string: フィールドはテキストボックスになります。STRING 型の値を受け入れます。
 - number: フィールドは数値入力ボックスになります。NUMBER タイプの値を受け入れます。
 - password: フィールドは値が非表示になるテキストボックスになります。STRING 型の値を受け入れます。
 - textarea: フィールドは幅の広いテキストボックスになります。STRING 型の値を受け入れます。
 - checkbox: このフィールドには BOOL 型の値を受け入れます。
 - select: このフィールドは、単一の値を選択するドロップダウンになります。VALUE 型の値を受け入れます。
 - multiselect: フィールドは、いくつかのオプションから選択できる複数選択ボックスになります。VALUE 型の値を受け入れます。
 - tree: フィールドはさまざまなレベルのツリーになり、その要素を選択して値を送信するかどうかを選択できます。VALUE 型の値を受け入れます。
- placeholder:
 - 関連付けられた type が string または textarea の場合はオプションです。
 - テキストボックスに例として表示されるテキストを定義します。
 - 使用されるキーは、このブロックのマクロ配列に存在する必要があります。
- show_on_true:
 - オプション。
 - フォーム上の他のチェックボックスフィールドのマクロを値として定義します。これにより、それらのマクロがフォーム上で true の値を持つ場合、目的のフィールドが表示されます。
 - 使用されるキーは、このブロックのマクロ配列に存在する必要があります。
- encrypt_on_true:
 - 関連付けられた type が password の場合はオプションです。
 - フォーム内の checkbox タイプの他のフィールドのマクロを値として定義します。これにより、これらのマクロがフォーム内で値 true を持つ場合、設定を保存するときに目的のフィールドが暗号化されます。
 - 暗号化/復号化は、discovery_extension_definition ブロックの設定に基づいて行われます。
 - 使用されるキーは、このブロックのマクロ配列に存在する必要があります。
- select_data:
 - 関連付けられた type が select または multiselect の場合は必須です。
 - ドロップダウンの値のソースを定義します。
 - セレクタは、Pandora FMS からのデータを使用して生成することも、カスタマイズすることもできます。これを行うために、次の値が受け入れられます。
 - select または multiselect タイプの要素が取得される、同じ INI ファイル内の他の設定ブロックの名前。
 - agent_groups: エージェントグループをデータとして使用します。
 - agents: エージェントをデータとして使用します。
 - module_groups: モジュールグループをデータとして使用します。
 - modules: モジュールをデータとして使用します。
 - module_types: モジュールタイプをデータとして使用します。
 - tags: タグをデータとして使用します。
 - status: ステータスをデータとして使用します。

- alert_templates: アラートテンプレートをデータとして使用します。
 - alert_actions: アラートアクションをデータとして使用します。
 - interval: 時間セクターをデータとして使用します。
 - credentials.custom: カスタム認証情報セクターをデータとして使用します。
 - credentials.aws: AWS 認証情報セクターをデータとして使用します。
 - credentials.azure: Azure 認証情報セクターをデータとして使用します。
 - credentials.gcp: Google Cloud 認証情報セクターをデータとして使用します。
 - credentials.sap: SAP 認証情報セクターをデータとして使用します。
 - credentials.snmp: SNMP 認証情報セクターをデータとして使用します。
 - credentials.wmi: WMI 認証情報セクターをデータとして使用します。
 - os: OSをデータとして利用します。
- 使用されるキーは、このブロックの マクロ 配列に存在する必要があります。
 - 関連付けられた type が multiselect の場合、マクロの値は、異なる値が選択された array タイプの JSON になります。 例 :

```
[1,5,12,23]
```

- tree_data:
 - 関連付けられた type が tree の場合は必須です。
 - tree タイプの要素が取得される同じ INI ファイルの設定ブロックを定義します。
 - 使用されるキーは、このブロックの マクロ 配列に存在する必要があります。
 - マクロの値は、さまざまな値が選択された array タイプの JSON になります。 例 :

```
["elementA", "elementF", "elementK"]
```

たとえば、これはいくつかの設定フィールドを定義する方法です。

```
[custom_fields_1]

macro[1] = _param1_
name[1] = User
type[1] = string

macro[2] = _param2_
name[2] = Password
type[2] = password
encrypt_on_true[2] = _param3_

macro[3] = _param3_
name[3] = Encrypt password
type[3] = checkbox

macro[4] = _param4_
name[4] = Max threads
type[4] = number
mandatory_field[4] = false

macro[5] = _param5_
name[5] = Agents group
tip[5] = Agents are generated in this group
type[5] = select
select_data[5] = agent_groups
```

```
macro[6] = _param6_
name[6] = Mode
type[6] = select
select_data[6] = custom_select_1

macro[7] = _param7_
name[7] = Add extra options
type[7] = checkbox

macro[8] = _param8_
name[8] = Extra elements
type[8] = tree
tree_data[8] = custom_tree_1
show_on_true[8] = _param7_

macro[9] = _param9_
name[9] = Extra options
type[9] = textarea
placeholder[9] = "Add extra options here"
show_on_true[9] = _param7_
```

次の例は、各ケースのコメントで説明されているすべてのパラメーターが含まれているため、このブロックのベースとして使用できます。

```
[CUSTOM_FIELDS_N]

; Mandatory
; Defines configuration field unique macro
; Macros can be used for script executions, using their value in place

macro[N] = _FIELD_MACRO_N_

; Optional
; Defines if configuration field is mandatory or not
; By default all configuration fields are mandatory

mandatory_field[N] = BOOL

; Mandatory
; Defines configuration field name to be displayed in console
; Macro will be used as name if this field is empty

name[N] = FIELD_NAME

; Optional
; Define a tip for the field, to be displayed in console

tip[N] = FIELD_TIP

; Mandatory
; Defines the field type to be displayed in console
```

```
; Possible values:
; string
; number
; password
; textarea
; check box
; select
; multi select
; tree

type[N] = FIELD_TYPE

; Optional if type is string or textarea
; Defines a placeholder for the field, to be displayed in console

placeholder[N] = PLACEHOLDER

; Optional
; Field is shown in console only if assigned field macro exists, is checkbox and
is true

show_on_true[N] = _FIELD_MACRO_N_

; Optional if type is password
; Field value is encrypted when stored into database if assigned field macro
exists, is checkbox and is true
; Password encrypt and decrypt depends on scripts uploaded to the discovery
application

encrypt_on_true[N] = _FIELD_MACRO_N_

; Mandatory if type is select or multiselect
; Defines select data to be displayed in console
; Possible values:
; agent_groups - Uses agent groups names
; agents - Uses agents names
; module_groups - Uses module groups
; modules - Uses modules names
; module_types - Uses module types names
; tags - Uses module tags
; status - Uses module status names
; alert_templates - Uses alert templates names
; alert_actions - Uses alert actions names
; interval - Uses time interval selector
; credentials.custom - Uses pandora custom credentials selector
; credentials.aws - Uses pandora AWS credentials selector
; credentials.azure - Uses pandora Microsoft Azure credentials selector
; credentials.gcp - Uses pandora Google Cloud Platform credentials selector
; credentials.sap - Uses pandora SAP credentials selector
; credentials.snmp - Uses pandora SNMP credentials selector
; credentials.wmi - Uses pandora WMI credentials selector
; os - Uses pandora OS names
```

```

; CUSTOM_SELECT_N - Uses custom selector values
; multiselect fields value is a comma separated list of selected values

select_data[N] = FIELD_DATA

; Mandatory if type is tree
; Defines tree data to be displayed in console
; tree fields value is a comma separated list of selected values

tree_data[N] = CUSTOM_TREE_DATA_N

```

CUSTOM_SELECT ブロック

このタイプのブロックには、select_data パラメーターの CUSTOM_FIELDS ブロック内の値として割り当てられている限り、任意の名前を付けることができます。これらはプラグイン設定フォームのドロップダウンのオプションを定義しており、次のパラメータがあります。

- option:
 - 必須。
 - カスタムセレクトのオプションを定義します。
 - このパラメーターは 配列 で、そのキーはカスタムセレクトオプションの値であり、その値はカスタムセレクトの各オプションに表示されるテキストです。
 - このパラメータは 配列であるため、異なるキーを使用して複数回指定できます。

例:

```

[custom_select_1]

option[v1] = Value 1
option[v2] = Value 2
option[v3] = Value 3
option[v4] = Value 4
option[v5] = Value 5

```

次の例は、各ケースのコメントで説明されているすべてのパラメーターが含まれているため、このブロックのベースとして使用できます。

```

[CUSTOM_SELECT_N]

; Mandatory
; Defines the value-text pair for the select or multiselect
; Several value-text pairs can be defined

option[VALUE_N] = TEXT_N

```

CUSTOM_TREE_DATA ブロック

このタイプのブロックには、tree_data パラメーターの CUSTOM_FIELDS ブロック内、またはその children パラメーターの別の CUSTOM_TREE_DATA ブロック内の値として割り当てられている限り、任意の名前を付けることができます。これらはプラグイン設定フォームのツリーの要素を定義し、次のパラメータがあります。

These rules apply for to all parameters of this block:

- パラメータは配列であり、そのキーは位置指定になります。つまり、数値で示すことも、示さないこともできます。キーを示すことをお勧めします。
- 配列であるため、パラメータは異なるキーを使用して複数回指定できます。
- キーを共有するすべてのパラメータは、同じ要素を参照します。つまり、(パラメータに応じて) 異なる値を同じ要素に割り当てます。
- name:
 - 必須。
 - ツリーのこのレベルのさまざまな要素の名前を定義します。
- selectable:
 - オプション。
 - ツリーのこのレベルの要素が選択可能かどうかを定義します。
 - デフォルトでは、すべてのツリー要素が選択可能です。
 - 使用されるキーは、このブロックの name 配列に存在する必要があります。
- macro:
 - 関連する selectable が true の場合はオプション。
 - フォームで選択されたツリーのこのレベルの要素の値が保存される マクロ を定義します。
 - 値は 有効なマクロ である必要があります。
 - 同じ マクロ を同じツリーの異なる要素に対して定義できます。
 - マクロは、ツリー外の他の要素のマクロと同じにすることはできません。
 - マクロが定義されておらず、関連する selectable が true の場合、ツリー要素の値はツリーに定義されたマクロに保存されます。
 - 使用されるキーは、このブロックの name 配列に存在する必要があります。
- value:
 - 関連する selectable が true の場合は必須。
 - ツリーのこのレベルのさまざまな要素の値を定義します。
 - 使用されるキーは、このブロックの name 配列に存在する必要があります。

1. マクロの値は、同じマクロを共有するツリー内で選択されたさまざまな値を含む、タイプ array の JSON になります。例：

```
["element1A", "element1F", "element1K"]
```

- children:
 - オプション。
 - ツリーのこのレベルの子要素が取得される同じ INI ファイル内の他の設定ブロックの名前を定義します。
 - 無限ループを避けるために、ツリーの上位レベルで使用されている同じ INI の設定ブロックを参

照することはできません。

たとえば、これはツリーのいくつかの要素を定義する方法です。

```
[custom_tree_1]

name[1] = Performance modules
selectable[1] = false
children[1] = custom_tree_1_A

name[2] = Counter modules
selectable[2] = false
children[2] = custom_tree_1_B

[custom_tree_1_A]

name[1] = Profile1
macro[1] = _param10_
value[1] = p1

name[2] = Perf2
macro[2] = _param10_
value[2] = p2

name[3] = Perf3
macro[3] = _param10_
value[3] = p3

[custom_tree_1_B]

name[1] = Counter1
macro[1] = _param11_
value[1] = c1

name[2] = Counter1
macro[2] = _param11_
value[2] = c2

name[3] = Counter1
macro[3] = _param11_
value[3] = c3
```

次の例は、各ケースのコメントで説明されているすべてのパラメーターが含まれているため、このブロックのベースとして使用できます。

```
[CUSTOM_TREE_DATA_N]

; Mandatory
; Defines the name for the tree element
; Several names can be defined
```

```
name[N] = TEXT_N

; Optional
; Defines if tree element is selectable or not
; By default all tree elements are selectable
; Several selectables can be defined

selectable[N] = VALUE_N

; Optional if selectable is true
; Defines the macro where value is stored for the tree element
; Several macros can be defined
; Same macro can be defined for several tree elements inside the same tree
; Macro can't be the same as other outside the tree
; If no macro is defined, value is stored for the global tree macro
; Tree values are stored and used the same way as multiselect values do

macro[N] = FIELD_MACRO_N

; Mandatory if selectable is true
; Defines the value for the tree element
; Several values can be defined

value[N] = VALUE_N

; Optional
; Defines the children elements for the tree element
; CUSTOM_TREE_DATA_M can't be the same than in an upper level
; Several children can be defined

children[N] = CUSTOM_TREE_DATA_M
```

Pandora FMS データを使用したセレクトク

フォームに表示されるフィールド内で、単純な複数のセレクトクにはPandora FMS アプリケーション自体からのデータを使用することができます。

必要なデータに応じて、保存される値と実行で使用される値は異なります。

- agent_groups:
 - セレクトクでは、エージェントグループの名前が使用されます。
 - IDをデータとして保存します。
 - 実行時にグループ ID が存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。
- agents:
 - セレクトクでは、エージェントの別名が使用されます。
 - 名前をデータとして保存します。
 - 実行時にエージェントが存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。
- module_groups:

- セレクターではモジュールグループを使用します。
- IDをデータとして保存します。
- 実行時にグループ ID が存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。
- modules:
 - セレクターではモジュール名が使用されます。
 - 名前をデータとして保存します。
 - 実行時に、その名前のモジュールが存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。
- module_types:
 - セレクターでは、モジュールタイプの説明を使用します。
 - 名前をデータとして保存します。
 - 実行では、それらの名前が値として使用されます。
- tags:
 - セレクターにはタグを使用します。
 - IDをデータとして保存します。
 - 実行時にタグ ID が存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。
- status:
 - セレクターには、モジュールの状態の名前が使用されます。
 - 定数をデータとして保存します。
 - 実行では、定数を値として使用します。
 - 0: 正常
 - 2: 警告
 - 1: 障害
 - 3: 不明
 - 5: 未初期化
- alert_templates:
 - セレクターには、モジュールテンプレートの名前が使用されます。
 - IDをデータとして保存します。
 - 実行時にテンプレート ID が存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。
- alert_actions:
 - セレクターには、アラートアクションの名前が使用されます。
 - IDをデータとして保存します。
 - 実行時にアクション ID が存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。
- interval:
 - セレクターには、間隔セレクターを使用します。
 - 時間を秒単位で保存します。
 - 実行では、秒単位の時間を値として使用します。
- credentials.custom:
 - セレクターには、カスタム認証情報セレクターが使用されます。
 - IDをデータとして保存します。
 - 実行時に、資格情報 ID が存在するかどうかを確認し、存在する場合は、資格情報データを値として含む JSON の Base64 を使用します。そうでない場合は空になります。
 - このタイプの認証情報の JSON は次の形式になります。

```
{  
  "user": "USER",  
  "password": "PASSWORD"  
}
```

- credentials.aws:

- セレクターには AWS 認証情報セレクターを使用します。
- IDをデータとして保存します。
- 実行時に、資格情報 ID が存在するかどうかを確認し、存在する場合は、資格情報データを値として含む JSON の Base64 を使用します。そうでない場合は空になります。
- このタイプの認証情報の JSON は次の形式になります。

```
{
  "access_key_id": "ACCESS_KEY_ID",
  "secret_access_key": "SECRET_ACCESS_KEY"
}
```

- credentials.azure:

- セレクターには Azure 認証情報セレクターを使用します。
- IDをデータとして保存します。
- 実行時に、資格情報 ID が存在するかどうかを確認し、存在する場合は、資格情報データを値として含む JSON の Base64 を使用します。そうでない場合は空になります。
- このタイプの認証情報の JSON は次の形式になります。

```
{
  "client_id": "CLIENT_ID",
  "application_secret": "APPLICATION_SECRET",
  "tenant_domain": "TENANT_DOMAIN",
  "subscription_id": "SUBSCRIPTION_ID"
}
```

- credentials.gcp:

- セレクターには Google Cloud Platform 認証情報セレクターを使用します。
- IDをデータとして保存します。
- 実行時に、資格情報 ID が存在するかどうかを確認し、存在する場合は、資格情報データを値として含む JSON の Base64 を使用します。そうでない場合は空になります。
- このタイプの認証情報の JSON は次の形式になります。

```
{
  "type": "service_account",
  "project_id": "PROJECT_ID",
  "private_key_id": "PRIVATE_KEY_ID",
  "private_key": "PRIVATE_KEY",
  "client_email": "CLIENT_EMAIL",
  "client_id": "CLIENT_ID",
  "auth_uri": "AUTH_URI",
  "token_uri": "TOKEN_URI",
  "auth_provider_x509_cert_url": "AUTH_PROVIDER_X509_CERT_URL",
  "client_x509_cert_url": "CLIENT_X509_CERT_URL"
}
```

- credentials.sap:

- セレクターには SAP 認証情報セレクターを使用します。
- IDをデータとして保存します。
- 実行時に、資格情報 ID が存在するかどうかを確認し、存在する場合は、資格情報データを値として含む JSON の Base64 を使用します。そうでない場合は空になります。
- このタイプの認証情報の JSON は次の形式になります。

```
{
  "user": "USER",
  "password": "PASSWORD"
}
```

- credentials.snmp:

- セレクターには「SNMP 認証情報セレクター」を使用します。
- IDをデータとして保存します。
- 実行時に、資格情報 ID が存在するかどうかを確認し、存在する場合は、資格情報データを値として含む JSON の Base64 を使用します。そうでない場合は空になります。
- このタイプの認証情報の JSON は次の形式になります。

```
{
  "community": "COMMUNITY",
  "version": "VERSION",
  "securityLevelV3": "SECURITY_LEVEL_V3",
  "authUserV3": "USER_AUTH_V3",
  "authMethodV3": "AUTH_METHOD_V3",
  "authPassV3": "AUTH_PASS_V3",
  "privacyMethodV3": "PRIVACY_METHOD_V3",
  "privacyPassV3": "PRIVACY_PASS_V3"
}
```

- credentials.wmi:

- セレクターには「WMI 認証情報セレクター」を使用します。
- IDをデータとして保存します。
- 実行時に、資格情報 ID が存在するかどうかを確認し、存在する場合は、資格情報データを値として含む JSON の Base64 を使用します。そうでない場合は空になります。
- このタイプの認証情報の JSON は次の形式になります。

```
{
  "user": "USER",
  "password": "PASSWORD",
  "namespace": "NAMESPACE"
}
```

- OS:

- セレクターでは OS の名前が使用されます。
- IDをデータとして保存します。
- 実行時に、OS ID が存在するかどうかを確認し、存在する場合はその名前を値として使用します。そうでない場合は空になります。

ベースファイル

次の例は、discovery_definition.inファイルを作成する際の実データとして使用できます。これには、各ケースのコメントとともに上で説明した、考えられるすべてのブロックとパラメーターが含まれています。

```
;-----;
; DISCOVERY APPLICATION INI FILE BASE ;
;-----;
```

```
; Mandatory
; Defines global application data

[discovery_extension_definition]

; Mandatory
; Defines discovery application short name
; Short name must be unique
; Short names with "pandorafms." prefixes are used by Pandora FMS for official
applications

short_name = DISCOVERY_APPLICATION_UNIQUE_NAME

; Mandatory
; Defines the section where application will be shown in console
; Possible values:
; apps
; clouds
; custom

section = app

; Mandatory
; Defines discovery application name, shown in console

name = DISCOVERY_APPLICATION_NAME

; Mandatory
; Defines discovery application version, shown in console

version = VERSION

; Optional
; Defines discovery application description, shown in console

description = DESCRIPTION

; Optional
; Defines execution files inside .disk
; Several execution files can be defined

execution_file[_EXEC_MACRO_N_] = SCRIPT.pl

; Mandatory
; Defines execution for discovery server
; Several executions can be defined
; At least 1 is required for server execution

exec[] = SERVER_EXECUTION

; Optional
; Define password encrypt script
```

```
passencrypt_script = PASS_ENCRYPT_SCRIPT.pl

; Optional
; Defines password encrypt script execution format
; _passencrypt_script_ is replaced with the passencrypt_script file path
; _password_ is replaced with the string (password) to encrypt

passencrypt_exec = EXECUTION

; Optional
; Define password decrypt script

passdecrypt_script = PASS_DECRYPT_SCRIPT.pl

; Optional
; Defines password encrypt script execution format
; _passdecrypt_script_ is replaced with the passdecrypt_script file path
; _password_ is replaced with the string (password) to encrypt

passdecrypt_exec = EXECUTION

; Optional
; Defines the default values for the fields when a new task is created
; By default all values are empty or not selected
; Several default values can be defined
; Possible values depending on field type:
; string - STRING
; number - NUMBER
; password-STRING
; textarea-STRING
; checkbox-BOOL
; select-VALUE_N
; multiselect - [VALUE_1,VALUE_N]
; tree - [VALUE_1,VALUE_N]

default_value[_FIELD_MACRO_N_] = DEFAULT_VALUE

;-----;

; Optional
; Defines application configuration steps

[config_steps]

; Following parameters can be setup for each configuration step
; Several steps can be defined using a different key (N)

; Mandatory
; Defines configuration step name

name[N] = STEP_NAME
```

```
; Mandatory if not custom_fields defined
; Defines configuration fields retrieved to console by a command execution
; execution_file scripts can be used to retrieve data
; Command execution output must be JSON as follows
; [
; {
; "macro": "_FIELD_MACRO_N_",
; "mandatory_field": "BOOL",
; "name": "FIELD_NAME",
; "tip": "FIELD_TIP",
; "type": "FIELD_TYPE",
; "placeholder": "PLACEHOLDER",
; "show_on_true": "_FIELD_MACRO_N_",
; "encrypt_on_true": "_FIELD_MACRO_N_",
; "select_data": {
; "VALUE_1": "TEXT_1",
; "VALUE_N": "TEXT_N"
; },
; "tree_data": [
; {
; "name": "TEXT_1",
; "selectable": "BOOL_1",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_1",
; "childin": [
; {
; "name": "TEXT_1_1",
; "selectable": "BOOL_1_1",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_1_1",
; "children": []
; },
; {
; "name": "TEXT_1_N",
; "selectable": "BOOL_1_N",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_1_N",
; "children": []
; }
; ]
; },
; {
; "name": "TEXT_N",
; "selectable": "BOOL_N",
; "macro": "_FIELD_MACRO_N_",
; "value": "VALUE_N",
; "children": []
; }
; ]
; }
; ]
```

```
script_data_fields[N] = CONSOLE_EXECUTION_FIELDS

; Mandatory if not script_data_fields defined
; Defines custom configuration fields

custom_fields[N] = CUSTOM_FIELDS_N

; Optional
; Defines the number of fields columns for the configuration steps (1 or 2)

fields_columns[N] = M

;-----;

; Mandatory if not script_data_fields defined
; Defines custom configuration fields to be used by configuration steps

[CUSTOM_FIELDS_N]

; Mandatory
; Defines configuration field unique macro
; Macros can be used for script executions, using their value in place

macro[N] = _FIELD_MACRO_N_

; Optional
; Defines if configuration field is mandatory or not
; By default all configuration fields are mandatory

mandatory_field[N] = BOOL

; Mandatory
; Defines configuration field name to be displayed in console
; Macro will be used as name if this field is empty

name[N] = FIELD_NAME

; Optional
; Define a tip for the field, to be displayed in console

tip[N] = FIELD_TIP

; Mandatory
; Defines the field type to be displayed in console
; Possible values:
; string
; number
; password
; textarea
; check box
; select
; multi select
```

```
; tree

type[N] = FIELD_TYPE

; Optional if type is string or textarea
; Defines a placeholder for the field, to be displayed in console

placeholder[N] = PLACEHOLDER

; Optional
; Field is shown in console only if assigned field macro exists, is checkbox and
is true

show_on_true[N] = _FIELD_MACRO_N_

; Optional if type is password
; Field value is encrypted when stored into database if assigned field macro
exists, is checkbox and is true
; Password encrypt and decrypt depends on scripts uploaded to the discovery
application

encrypt_on_true[N] = _FIELD_MACRO_N_

; Mandatory if type is select or multiselect
; Defines select data to be displayed in console
; Possible values:
; agent_groups - Uses agent groups names
; agents - Uses agents names
; module_groups - Uses module groups
; modules - Uses modules names
; module_types - Uses module types names
; tags - Uses module tags
; status - Uses module status names
; alert_templates - Uses alert templates names
; alert_actions - Uses alert actions names
; interval - Uses time interval selector
; credentials.custom - Uses pandora custom credentials selector
; credentials.aws - Uses pandora AWS credentials selector
; credentials.azure - Uses pandora Microsoft Azure credentials selector
; credentials.gcp - Uses pandora Google Cloud Platform credentials selector
; credentials.sap - Uses pandora SAP credentials selector
; credentials.snmp - Uses pandora SNMP credentials selector
; credentials.wmi - Uses pandora WMI credentials selector
; os - Uses pandora OS names
; CUSTOM_SELECT_N - Uses custom selector values
; multiselect fields value is a comma separated list of selected values

select_data[N] = FIELD_DATA

; Mandatory if type is tree
; Defines tree data to be displayed in console
; tree fields value is a comma separated list of selected values
```

```
tree_data[N] = CUSTOM_TREE_DATA_N

;-----;

; Mandatory if custom tree defined
; Defines custom tree values to be used by configuration fields

[CUSTOM_TREE_DATA_N]

; Mandatory
; Defines the name for the tree element
; Several names can be defined

name[N] = TEXT_N

; Optional
; Defines if tree element is selectable or not
; By default all tree elements are selectable
; Several selectables can be defined

selectable[N] = VALUE_N

; Optional if selectable is true
; Defines the macro where value is stored for the tree element
; Several macros can be defined
; Same macro can be defined for several tree elements inside the same tree
; Macro can't be the same as other outside the tree
; If no macro is defined, value is stored for the global tree macro
; Tree values are stored and used the same way as multiselect values do

macro[N] = FIELD_MACRO_N

; Mandatory if selectable is true
; Defines the value for the tree element
; Several values can be defined

value[N] = VALUE_N

; Optional
; Defines the children elements for the tree element
; CUSTOM_TREE_DATA_M can't be the same than in an upper level
; Several children can be defined

children[N] = CUSTOM_TREE_DATA_M

;-----;

; Mandatory if custom select or multiselect defined
; Defines custom select or multiselect values to be used by configuration fields

[CUSTOM_SELECT_N]
```

```
; Mandatory
; Defines the value-text pair for the select or multiselect
; Several value-text pairs can be defined

option[VALUE_N] = TEXT_N

;-----;

; Optional
; Defines temporary configuration files to be created and used during script
executions

[tempfile_confs]

; Mandatory
; Defines the content for the temporary file
; File will be used where temporary file macro is specified during executions
; File content replaces fields macros with their values

file[_TEMP_FILE_MACRO_N_] = _FIELD_MACRO_N_,_FIELD_MACRO_M_
```

例

以下に、discovery_definition.ini ファイルの例と、そのフォームが Pandora FMS コンソールにどのように表示されるかを示します。

```
[discovery_extension_definition]

short_name = discoveryTest
section = custom
name = Discovery test
version="1.0"
description = A test discovery plugin

execution_file[_exec1_] = test.py

exec[] = "'_exec1_' -c '_tempConf_'"

passencrypt_script = pass_encrypter.py
passencrypt_exec = "'_passencrypt_script_' --encrypt '_password_'"

passdecrypt_script = pass_encrypter.py
passdecrypt_exec = "'_passdecrypt_script_' --decrypt '_password_'"

default_value[_param1_] = "admin"
default_value[_param2_] = ""
default_value[_param3_] = false
default_value[_param4_] = 5
default_value[_param5_] = 0
default_value[_param6_] = v1
```

```
default_value[_param7_] = true
default_value[_param8_] = "[]"
default_value[_param9_] = ""
default_value[_param10_] = "[]"
default_value[_param11_] = "[]"

[config_steps]

name[1] = Configuration step
custom_fields[1] = custom_fields_1

[tempfile_confs]

file[_tempConf_] = "user _param1_
password _param2_
encrypt_pass _param3_

threads _param4_

group _param5_

mode _param6_
extra_options _param7_
extra_elements _param8_
extra_perfs _param10_
extra_counters _param11_

log __temp__/__taskMD5__.log

_param9_"

[custom_fields_1]

macro[1] = _param1_
name[1] = User
type[1] = string

macro[2] = _param2_
name[2] = Password
type[2] = password
encrypt_on_true[2] = _param3_

macro[3] = _param3_
name[3] = Encrypt password
type[3] = checkbox

macro[4] = _param4_
name[4] = Max threads
type[4] = number
mandatory_field[4] = false

macro[5] = _param5_
```

```
name[5] = Agents group
tip[5] = Agents are generated in this group
type[5] = select
select_data[5] = agent_groups

macro[6] = _param6_
name[6] = Mode
type[6] = select
select_data[6] = custom_select_1

macro[7] = _param7_
name[7] = Add extra options
type[7] = checkbox

macro[8] = _param8_
name[8] = Extra elements
type[8] = tree
tree_data[8] = custom_tree_1
show_on_true[8] = _param7_

macro[9] = _param9_
name[9] = Extra options
type[9] = textarea
placeholder[9] = "Add extra options here"
show_on_true[9] = _param7_

[custom_select_1]

option[v1] = Value 1
option[v2] = Value 2
option[v3] = Value 3
option[v4] = Value 4
option[v5] = Value 5

[custom_tree_1]

name[1] = Performance modules
selectable[1] = false
children[1] = custom_tree_1_A

name[2] = Counter modules
selectable[2] = false
children[2] = custom_tree_1_B

[custom_tree_1_A]

name[1] = Profile1
selectable[1] = true
macro[1] = _param10_
value[1] = p1

name[2] = Perf2
```

```
selectable[2] = true
macro[2] = _param10_
value[2] = p2

name[3] = Perf3
selectable[3] = true
macro[3] = _param10_
value[3] = p3

[custom_tree_1_B]

name[1] = Counter1
selectable[1] = true
macro[1] = _param11_
value[1] = c1

name[2] = Counter1
selectable[2] = true
macro[2] = _param11_
value[2] = c2

name[3] = Counter1
selectable[3] = true
macro[3] = _param11_
value[3] = c3
```

上記の定義では、アプリケーションのタスクを作成するときに、次のフォームが表示されます。

Discovery / Custom / Task definition / Configuration step

Discovery test

| | |
|-------------------------------------|--|
| User | Password |
| <input type="text" value="admin"/> | <input type="password"/> |
| Encrypt password | Max threads |
| <input type="checkbox"/> | <input type="text" value="5"/> |
| Agents group ⓘ | Mode |
| <input type="text" value="All"/> | <input type="text" value="Valor 1"/> |
| Add extra options | Extra elements |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Performance modules |
| | <input type="checkbox"/> Perf1 |
| | <input type="checkbox"/> Perf2 |
| | <input type="checkbox"/> Perf3 |
| | <input type="checkbox"/> Counter modules |
| | <input type="checkbox"/> Counter1 |
| | <input type="checkbox"/> Counter1 |
| | <input type="checkbox"/> Counter1 |
| Extra options | |
| <input type="text"/> | |

“追加オプションを追加(Add extra options)”が選択されていない場合、“追加要素(extra elements)”フィールドと“追加オプション(Extra options)”フィールドは非表示になり、“パスワード>Password)”フィールドは“暗号化パスワード(Encrypt password)”フィールドを選択することで値を暗号化します。

Discovery / Custom / Task definition / Configuration step

Discovery test

| | | | |
|-------------------|-------------------------------------|-------------|---------|
| User | admin | Password | |
| Encrypt password | <input checked="" type="checkbox"/> | Max threads | 5 |
| Agents group | All | Mode | Valor 1 |
| Add extra options | <input type="checkbox"/> | | |

このタスクがサーバによって実行されると、次のような内容の一時ファイル (例: /var/spool/pandora/data_in/discovery/tmp/6d7fce9fee471194aa8b5b6e47267f03) が生成されます。

```
user admin
password MjZhYjBkYjkwZDcyZTI4YWQwYmExZTIyZWU1MTA1MTAgIC0K
encrypt_pass 1

thread 2

group Applications

mode v1
extra_options 1
extra_elements[]
extra_perfs["p1", "p2"]
extra_counters["c1"]

log /tmp/b026324c6904b2a9cb4b88d6d61c81d1.log
```

そして、次のように実行中にこれが使用されます。

```
'/var/spool/pandora/data_in/discovery/discoveryTest/test.py' -c
'/var/spool/pandora/data_in/discovery/tmp/6d7fce9fee471194aa8b5b6e47267f03'
```

スクリプトと実行ファイル

Pandora FMS サーバは、`discovery_setting.ini` ファイルで定義されたスクリプトと実行可能ファイルの実行を担当し、各実行結果の出力とエラーコードに基づいて各タスクの実行結果と概要

を決定します。

ステータスを判断するために、それぞれの実行結果によって返されるエラーコードをチェックします。少なくとも 1 つの実行で 0 以外のエラーコードが返された場合、タスクのステータスは失敗とみなされます。

タスクの概要を表示するために Pandora FMS サーバは実行ごとに、標準出力 (STDOUT) とエラー出力 (STDERR) の両方の出力を収集します。通常、次の最小限の JSON 形式での出力があります。

```
{
  "summary": {
    "SUMMARY_FIELD_1": "SUMMARY_VALUE_1",
    "SUMMARY_FIELD_N": "SUMMARY_VALUE_N"
  },
  "info": "ADDITIONAL_INFO"
}
```

コンソールでは、summary キーは 2 列のテーブルとして読み取られ、そのキーは左側の要素、その値は右側の要素と見なされます。

info キーは追加情報として読み取られ、要約テーブルにさらに 1 行追加されます。

JSON の他のキー、または JSON が返されない場合、またはその構造に準拠していない場合は、残りの情報が概要テーブルのもう 1 行に追加されます。

実行ごとに概要テーブルが生成され、サーバーによって実行された実行順序に基づいてそれらガリストされます。このように、“概要 1” テーブルは最初の実行に対応し、“概要 2” テーブルは 2 番目の実行に対応し、以下同様になります。

これは、最後の実行の結果が、成功したか失敗したかにかかわらず、常にコンソールに表示されるようにすることを目的としています。

最後に、スクリプトまたは実行可能ファイルによって実行されるアクションとは別に、Pandora FMS サーバは実行の出力からエージェントとモジュールを処理できるようになります。このため Pandora FMS サーバは、処理する必要がある各エージェントとモジュールの情報を含む追加のキー monitoring_data を JSON 出力で受け取ることを期待しますが、このキーのデータは実行の概要には含まれません。

したがって、自動検出プラグインの予想される JSON 出力形式は次のようになります。

```
{
  "summary": {
    "SUMMARY_FIELD_1": "SUMMARY_VALUE_1",
    "SUMMARY_FIELD_N": "SUMMARY_VALUE_N"
  },
  "info": "ADDITIONAL_INFO",
  "monitoring_data": [
```

```
{
  "agent_data": {
    "agent_name": "AGENT_NAME",
    "agent_alias": "AGENT_ALIAS",
    "Bears",
    "os_version": "OS_VERSION",
    "interval": "INTERVAL",
    "id_group": "ID_GROUP",
    "address": "ADDRESS",
    "description": "DESCRIPTION",
    "agent_version": "AGENT_VERSION",
    "parent_agent_name": "PARENT_AGENT_NAME",
    "timezone_offset": "TIMEZONE_OFFSET"
  },
  "module_data": [
    {
      "name": "NAME",
      "data": "DATA",
      "type": "TYPE",
      "description": "DESCRIPTION",
      "max": "MAX",
      "min": "MIN",
      "post_process": "POST_PROCESS",
      "module_interval": "MODULE_INTERVAL",
      "min_critical": "MIN_CRITICAL",
      "max_critical": "MAX_CRITICAL",
      "min_warning": "MIN_WARNING",
      "max_warning": "MAX_WARNING",
      "disabled": "DISABLED",
      "min_ff_event": "MIN_FF_EVENT",
      "datalist": "DATALIST",
      "status": "STATUS",
      "unit": "UNIT",
      "timestamp": "TIMESTAMP",
      "module_group": "MODULE_GROUP",
      "custom_id": "CUSTOM_ID",
      "str_warning": "STR_WARNING",
      "str_critical": "STR_CRITICAL",
      "critical_instructions": "CRITICAL_INSTRUCTIONS",
      "warning_instructions": "WARNING_INSTRUCTIONS",
      "unknown_instructions": "UNKNOWN_INSTRUCTIONS",
      "tags": "TAGS",
      "critical_inverse": "CRITICAL_INVERSE",
      "warning_inverse": "WARNING_INVERSE",
      "quiet": "QUIET",
      "module_ff_interval": "MODULE_FF_INTERVAL",
      "alert_template": "ALERT_TEMPLATE",
      "crontab": "CRONTAB",
      "min_ff_event_normal": "MIN_FF_EVENT_NORMAL",
      "min_ff_event_warning": "MIN_FF_EVENT_WARNING",
      "min_ff_event_critical": "MIN_FF_EVENT_CRITICAL",
      "ff_timeout": "FF_TIMEOUT",
    }
  ]
}
```

```
"each_ff": "EACH_FF",
"module_parent": "MODULE_PARENT",
"module_parent_unlink": "MODULE_PARENT_UNLINK",
"cron_interval": "CRON_INTERVAL",
"ff_type": "FF_TYPE",
"min_warning_forced": "MIN_WARNING_FORCED",
"max_warning_forced": "MAX_WARNING_FORCED",
"min_critical_forced": "MIN_CRITICAL_FORCED",
"max_critical_forced": "MAX_CRITICAL_FORCED",
"str_warning_forced": "STR_WARNING_FORCED",
"str_critical_forced": "STR_CRITICAL_FORCED"
    }
  ]
}
}
```