Proxedo API Security: Release notes for PAS 4.12.1

The following new features, bug fixes and improvements have been completed for Release 4.12.1 Proxedo API Security.

Features

- **OpenID Connect authentication support**
  The OpenID Connect authentication method is now supported on *Endpoints*. The OpenID Connect identity provider can be configured through an *Authentication* brick.

- **Error Selectors and Matchers**
  New *Selectors* and *Matchers* are available for handling *Plugin* names, verdicts and error messages, along with *Error Policy* data from the selected *Plugin*. *Plugins* can be referred explicitly, or dynamically using the special *<Previous>* value.

- **Configurable response headers for Error Policy**
  The *Error Policy* components now have an *Error Response Headers* field. Headers configured here will be added to the error response for the client when an *Error Policy* fires. This can be used to add CORS headers to error responses.

- **Easy enable/disable option for Endpoints**
  Endpoints now have a field, *Traffic Mode*, that can be used to disable the *Endpoint* while keeping its configuration. Disabling can mean rejecting all traffic or permitting all traffic, depending on the use case.

- **Details are visible for edited components**
  The difference between the edited and the running components can be viewed on the Web UI's Changes page. The same difference can be also queried from the API.

- **Service version is visible on Status page**
  The version (container tag) of services is now visible on the Status page on the Web UI.

Bug Fixes

- **Header-related memory leak**
  A memory leak could lead to crashes after heavy traffic with many headers. This has been corrected.

- **Status Class comparators work more reliably**
  Status Class comparators used in matchers failed to match properly in some configurations. This has been corrected.

- **Apply Configuration is more stable on Kubernetes**
  Applying the configuration could fail in some cases on Kubernetes. This has been corrected.

- **URI Port matcher had wrong comparators**
  The *URI Port* matcher has string-based comparators instead of number-based ones. The comparator list has been corrected.

- **Some fields on the Web UI could not be reset**
  The values of some fields, like *Verbosity* in *Log* could not be reset to its original empty value. This has been corrected.

- **Edited component could be stuck in Changes**
  After editing a component and then undoing the changes manually would leave the component in the user's Changes without a difference between the edited and the original instance. The edited instance is not properly removed if necessary.

Improvements

- **Configuration Import errors**
  Errors occurring during importing a configuration backup are now visible on the Web UI.

- **Detailed validation errors for XSD and WSDL schemas**

When an invalid XSD or WSDL schema *File Brick* is uploaded the validation now produces more detailed error messages.

- **The original file names are now preserved**
  The original names of uploaded *File Bricks* are now preserved and used when downloading through the API, or when the configuration is exported.

- **Clearer placeholders in UI fields**
  The placeholder texts that appear in empty fields on the Web UI have been made more uniform and unambiguous.

- **CA renamed to CA Bundle**
  CA Directory fields and the *CA File Brick* type have been renamed to *CA Bundle* to be more consistent with industry standards.

- **Message field can be included in Elastic Insight Targets**
  The *Message* field of *Insight* plugins was not present in the output of *Elastic Insight Targets*. *Elastic Insight Targets* now include an *Include Message* flag and a *Message Key* field that specifies the key to use when assembling JSON output, similarly to *Syslog Insight Targets*.

- **More secure containers**
  Most services within the Docker containers run with non-root privileges, following the principle of least privilege. This enhances security by limiting access to the host system, even if a service is exposed to risk.

- **Shorter default port ranges for transport-director**
  The default port ranges for transport-director have been changed from 49000-49100 and 49101-49200 to 49000-49010 and 49011-49020. Most installations don't need 200 ports, and service restart times significantly benefit from shorter port ranges.

> ℹ️ Existing port ranges will not be changed by the upgrade process. If these ports are unused, ranges can be manually changed in */opt/balasys/etc/infrastructure/pas/docker-compose.conf*.

- **More consistent logging in flow-director**
  The component-related logs in flow-director are now more consistent and contain more information.

- **Applying the configuration returns proper status code**
  Previously, when applying the configuration failed, the API returned HTTP 412 (Precondition Failed). Now it returns a more appropriate HTTP 409 (Conflict).

- **Password generation now happens during installation**
  Previously the initial password was generated during the first start of the *config-api* service, unless a password has been specified by the user. Now this happens during installation, so the password store has been made read-only for added security.

- **The LDAP handling library in the config-api service has been replaced**
  The library that handles the LDAP protocol in the *config-api* has been replaced with a more performant solution.

- **The blob-store service has been replaced**
  The internal implementation of the *blob-store* service has been replaced from MinIO to Garage, a more modern and robust solution.

- **Support for new Docker versions**
  The `docker-compose` command has been deprecated in recent Docker version, so PAS now supports the `docker compose` format.