API Builder Tools 3.2.5 Release Note

API Builder 3.x is deprecated
Support for API Builder 3.x will cease on 30 April 2020. Use the v3 to v4 upgrade guide to migrate all your applications to API Builder 4.x.

Contact support@axway.com if you require migration assistance.

API Builder Tools 3.2.5 - 5 December 2018

API Builder Tools 3.2.5 is a minor release that includes a security fix, new features, improvements, and bug fixes.

As of this release, API Builder Tools 3.1.x will not be supported six months (2019-05-05) from 3.2.5's release date. See Axway Appcelerator Deprecation Policy and Nominal Lifetimes documents for details.

Fixed security issue

- Previously, an application built with API Builder could allow a remote attacker to bypass authentication to an API endpoint. The issue was discovered by Axway and we have no indications that the vulnerability has been exploited or publicly disclosed. Now, the authentication bypass vulnerability has been resolved in API Builder 3.x.
    - **Summary**: An application built with API Builder could allow a remote attacker to bypass authentication to an API endpoint. The issue was discovered by Axway and we have no indications that the vulnerability has been exploited or publicly disclosed.
    - **Vulnerability details**: An application built with API Builder could allow a remote attacker to bypass authentication to an API endpoint and obtain sensitive information, affect the integrity and availability when using the following authentication mechanisms:
        - HTTP Basic Authentication
        - API Key based Authentication
        - LDAP Authentication
    - **CVSS base score**: 10
    - **CVSS vector**: (CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:N)
    - **Applying remediation/fixes** : If your published applications are using any of the following authentication mechanisms, it is recommended that you download the patch versions and republish your applications:
        - HTTP Basic Authentication
        - API Key based Authentication
        - LDAP Authentication

      For API Builder Applications developed and published with **API Builder 3.x**, download CLI Version 7.0.7 or above and republish the application with the following steps:
        1. **appc use latest**
        2. **cd <your project folder>**
        3. **appc publish -f**

      If you have manually added the Arrow dependency to your API Builder application where the Arrow or API Builder version is a custom build, then ensure that this dependency is updated to the latest version (Arrow/API Builder 3.2.5 or above).
    - **Further information**: For additional questions, contact support@axway.com.

New features

- Added the ability to validate generated flows using the Axway Flow SDK
- Implemented selector auto-complete when creating and editing flows in the API Builder UI

Improvements

- Previously, the initial project created by the `appc new` command contained unnecessary dependencies. Now, the initial project created by the `appc new` command has been updated to remove unnecessary dependencies and to provide extra examples on how to test endpoints and models. Additionally, the `grunt` command is no longer used for the build. The `npm test` command is used instead. Also, the `example.md` web route has been removed.
- Previously, when editing flows, it was not possible to see previously used and available selectors. Now, when editing a selector any previously used selector, or any available selector will display a context-assist drop-down menu showing selector or selectors that match the input text.
- Previously, the `axway-flow-sdk` would generate projects that required transpiling with babel before they could be used. Now, the generated projects do not require transpiling; so the babel dependency has been removed, but the generated projects require a Node.js version equal to or greater than 8.9.x instead.
- Previously, the API SDK generator was used to generate software development kits based on integrated SDK templates. Now, the API Builder SDK generator has been deprecated and the API Builder application exposes its APIs definitions using a standard Swagger

format. These API definitions can be consumed by third-party SDK generators to create clients.

Fixed issues

- Previously, attempting to develop flow-nodes locally in the ./nodes directory would cause API Builder to also attempt to read the `node_modules` folder and cause the application to crash. Now, locally developed flow-node dependencies in the `./nodes/node_modules` folder will be ignored and any flow-node dependency that is required must be added and resolved via the API Builder application's `package.json` instead.
- Previously, API Builder would not handle JSON schema references on startup and could generate invalid Swagger definitions that were missing the global schema references. Now, API Builder will load and handle the JSON schema references as expected.
- Previously, the **appc generate** command would generate excessive logging. Now, the command has been reduced to avoid excessive logging. If required, the logging level can be increased with the `appc generate -l debug` option.
- Previously, `jsonselect@0.4.0` was used indirectly by `@axway/api-builder-runtime` and had three CVEs against it: CVE-2011-4969 - XSS with location.hash, CVE-2012-6708 - Selector interpreted as HTML, and CVE-2015-9251 - 3rd party CORS request may execute. Now, this module is no longer a runtime dependency.
- Previously, the **appc publish** command did not unpublish the oldest version of an application if more than 10 versions of the application were published. Now, when the **appc publish** command is executed and more than 10 versions of the application have been published, the oldest version of the application is unpublished.

Known issues

- The legacy Distinct() APIs have inconsistent or non-compliant return types.
- Clicking on joined models in the Source column on the **Models** tab of the API Builder Console does not work. The details of the joined model should be displayed.
- If a Swagger definition uses the `allOf` parameter, the body generated in the method test window is incorrect.
- Filtering the API Builder Console administrator access using IPv6 addresses may cause ENOTFOUND errors.
- When attempting to create and save a flow for an imported Swagger endpoint that contains a path or paths defined by references such as `GET /find`, a Page Not Found (404) error will be displayed in the API Builder Console and the flow cannot be saved. For example, a Swagger document with a path reference may look similar to this:

```
{
 "swagger": "2.0",
 "paths" {
  "x-path": {
   "get": {}
  },
  "/find": {
   "$ref": "#/paths/x-path"
  }
 }
}
```

  Currently, Swagger documents with path references can be imported and the imported endpoint will be correctly displayed in API Lists, but a flow cannot be created and saved successfully. When the flow editor **Save** button is clicked, a Page Not Found error is displayed. The error occurs because the API Builder Console cannot find the method as it is not in an expected location.
- When rendering the flow editor, the API Builder Console may fail to render the Scalable Vector Graphics (SVG) icons correctly in the Firefox browser. The render failure may result in blank icons being displayed in the tool panel and in the flow diagram. This is due to a long-standing bug in Firefox that fails to scale SVG graphics correctly. To fix, edit the SVG icon and add height and width. For example:

```
<svg ... height="80" width="80" />
```

- The API Builder Console does not recognize a required consumes value for form parameters if it is appended and the endpoint load will fail. For example:

```
"consumes": [ "multipart/form-data; charset=utf-8" ],
```

  The appended character set (`charset=utf-8`) will cause the endpoint load to fail.
- When deleting endpoints which contain references within paths, a Page Not Found (404) error may be displayed in the API Builder Console. For example, a Swagger document with references within paths may look similar to this:

```
{ "swagger": "2.0", "paths" { "x-path": { "get": {} }, "/find": { "$ref":
"#/paths/x-path" }, "/search": { "$ref": "#/paths/x-path" } } }
```

The API Builder Console will fail to find GET /find since it is inside a `$ref`. If the API Builder Console has modified the referenced `$ref`, it could cause unexpected behavior for other paths referencing #/paths/x-path such as /search - deleting GET /find could unexpectedly delete GET /search too.

- Endpoints named with special characters cannot be viewed in the API Builder Console. For example, if you attempt to view the `[test].json` endpoint, you will receive a Page Not Found error message.
- Editing large object parameters on the API Orchestration page in the API Builder Console may cause multiple, confusing node configuration panel scrollbars to appear.
- The inline object editor **Expand** button obscures text.
- When editing flows in the console log in the terminal, multiple lines with an unknown format are ignored and the following message is received: unknown format "multiline" ignored in the schema at path "#"
- In the flow editor, if a flow-node method description is too long, it may not be visible in its entirety.
- Editing large object parameters on the *API Orchestration* page in the API Builder Console may cause multiple, confusing flow-node configuration panel scrollbars to appear.

Security vulnerability

- bootstrap 20184
  - **Vulnerability**: API Builder UI uses react-bootstrap which as a dependency on bootstrap@3.3.7, and it is 3.3.7 that contains the vulnerability. At the time of writing this ticket, bootstrap did not publish a 3.x version after 3.3.7, so no fix was ever released. The bootstrap library is now at 4.x. However, the react-bootstrap library that API Builder uses is actively working on a version compatible with bootstrap v4.x. The following issue is logged against bootstrap@3.3.7:
    - 20184 - XSS in data-target attribute
  - **Analysis**: The vulnerability and risk are documented 20184 . The API Builder UI only runs on the developer machine and is locked to localhost by default. The API Builder UI **will not be installed in production**. Furthermore, the UI bundled with API Builder does not use data-target attributes. The risk to API Builder is low.