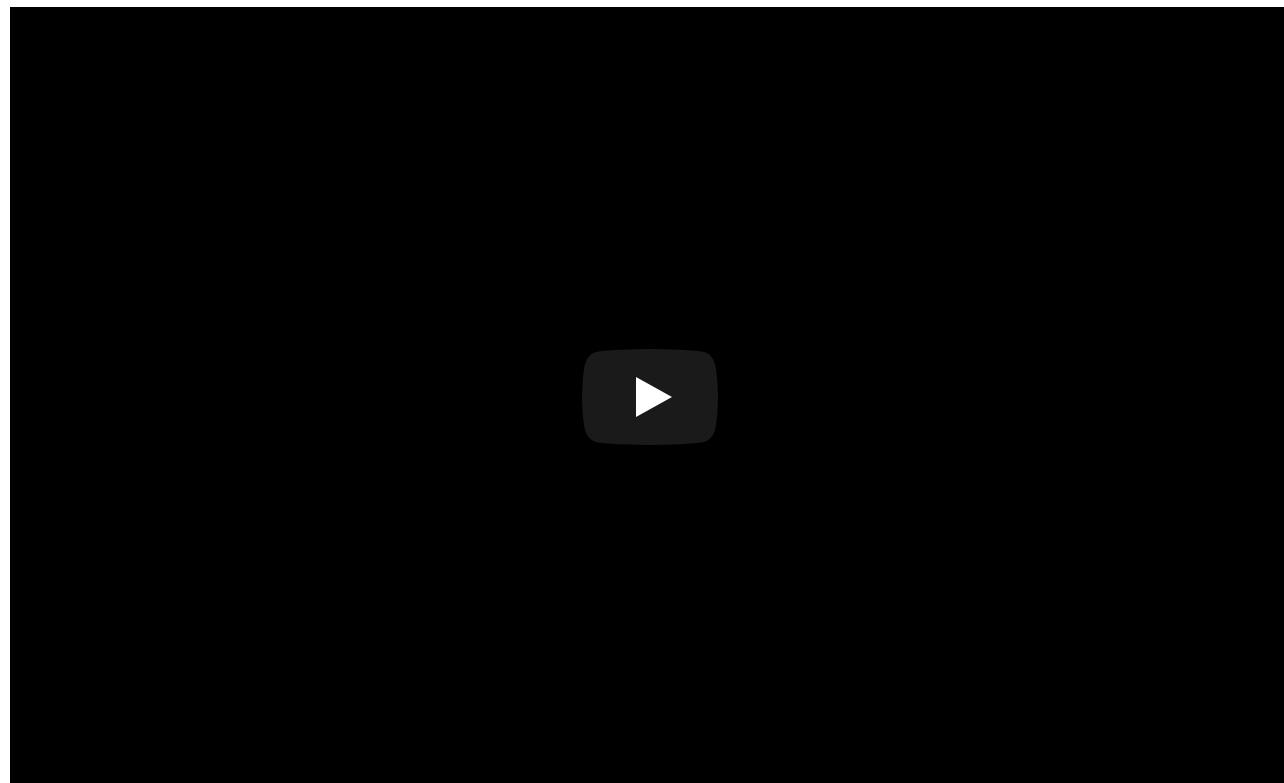


Zowe overview

Zowe™ is an open source software framework that allows mainframe development and operation teams to securely manage, control, script, and develop on the mainframe. It was created to host technologies that benefit the IBM Z platform for all members of the Z community, including Integrated Software Vendors (ISVs), System Integrators, and z/OS consumers. Like Mac or Windows, Zowe comes with a set of APIs and OS capabilities that applications build on and also includes some applications out of the box. Zowe offers modern interfaces to interact with z/OS and allows you to work with z/OS in a way that is similar to what you experience on cloud platforms today. You can use these interfaces as delivered or through plug-ins and extensions that are created by clients or third-party vendors. Zowe is a project within the Open Mainframe Project.

Zowe Demo Video

Watch this [video](#) to see a quick demo of Zowe.



[Download the deck for this video](#) | [Download the script](#)

Component Overview

Zowe consists of the following components:

Zowe Application Framework

A web user interface (UI) that provides a virtual desktop containing a number of apps allowing access to z/OS function. Base Zowe includes apps for traditional access such as a 3270 terminal and a VT Terminal, as well as an editor and explorers for working with JES, MVS Data Sets and Unix System Services.

 Learn more

z/OS Services

Provides a range of APIs for the management of z/OS JES jobs and MVS data set services.

 Learn more

Zowe CLI

Zowe CLI is a command-line interface that lets you interact with the mainframe in a familiar, off-platform format. Zowe CLI helps to increase overall productivity, reduce the learning curve for developing mainframe applications, and exploit the ease-of-use of off-platform tools. Zowe CLI lets you use common tools such as Integrated Development Environments (IDEs), shell commands, bash scripts, and build tools for mainframe development. Though its ecosystem of plug-ins, you can automate actions on systems such as IBM Db2, IBM CICS, and more. It provides a set of utilities and services for users that want to become efficient in supporting and building z/OS applications quickly.

 Learn more

Zowe Client Software Development Kits (SDKs)

The Zowe Client SDKs consist of programmatic APIs that you can use to build client applications or scripts that interact with z/OS. The following SDKs are available:

- Zowe Node.js Client SDK

- Zowe Python Client SDK

For more information, see [Using the Zowe SDKs](#).

API Mediation Layer

Provides a gateway that acts as a reverse proxy for z/OS services, together with a catalog of REST APIs and a dynamic discovery capability. Base Zowe provides core services for working with MVS Data Sets, JES, as well as working with z/OSMF REST APIs. The API Mediation Layer also provides a framework for [Single Sign On \(SSO\)](#).

 Learn more

To learn more about the architecture of Zowe, see [Zowe architecture](#).

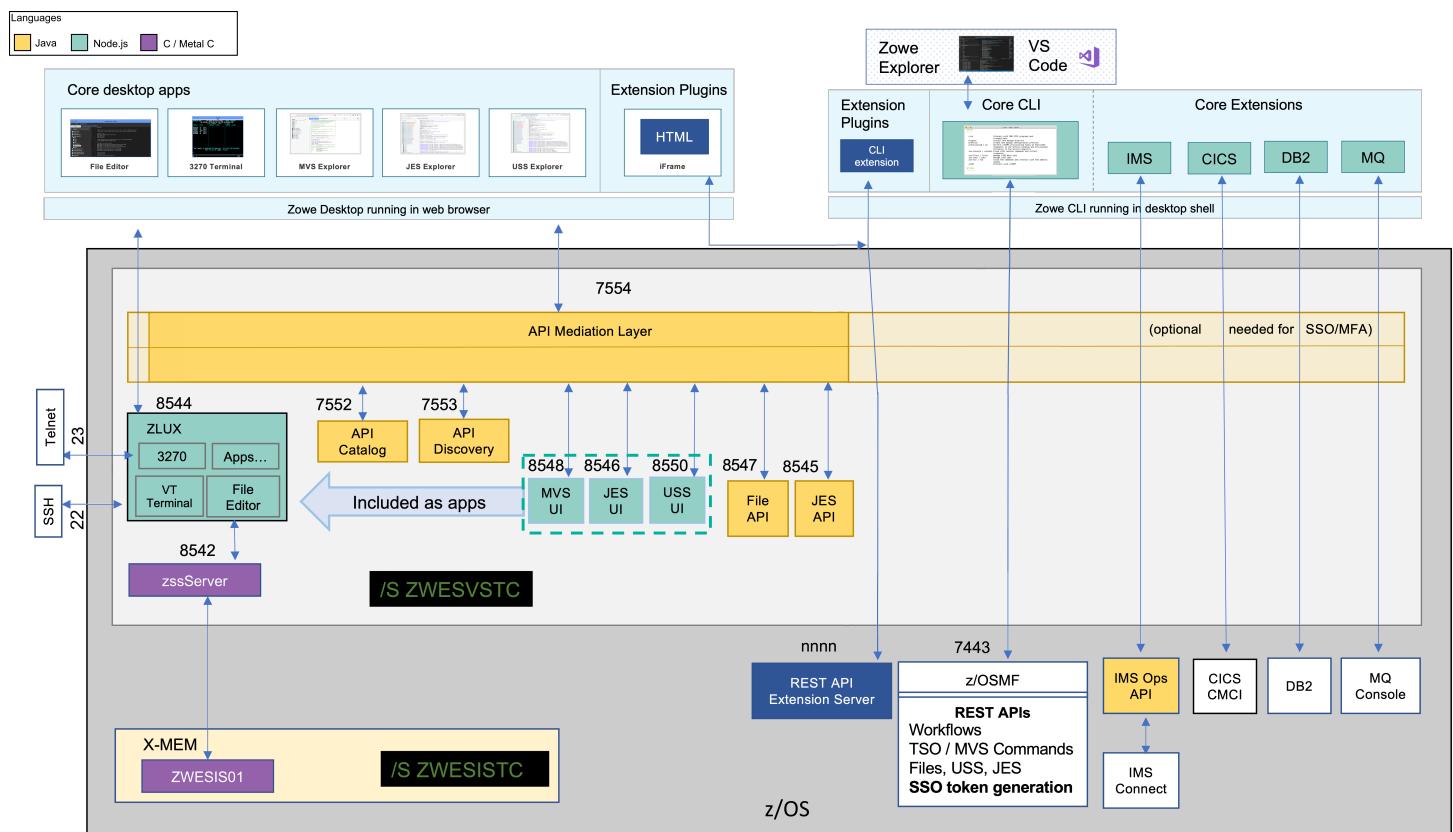
Zowe Third-Party Software Requirements and Bill of Materials

- [Third-Party Software Requirements \(TPSR\)](#)
- [Bill of Materials \(BOM\)](#)

Zowe architecture

Zowe™ is a collection of components that together form a framework that allows Z-based functionality to be accessible across an organization. This includes exposing Z-based components such as z/OSMF as Rest APIs. The framework provides an environment where other components can be included and exposed to a broader non-Z based audience.

The following diagram depicts the high-level Zowe architecture.



The diagram shows the default port numbers that are used by Zowe. These are dependent on each instance of Zowe and are held in the Zowe instance directory configuration file `instance.env`. For more information, see [Creating and configuring the Zowe instance directory](#).

Zowe components can be categorized by location: server or client. While the client is always an end user tool such as a PC, browser or mobile device, the server components can be further categorized by what machine they run on.

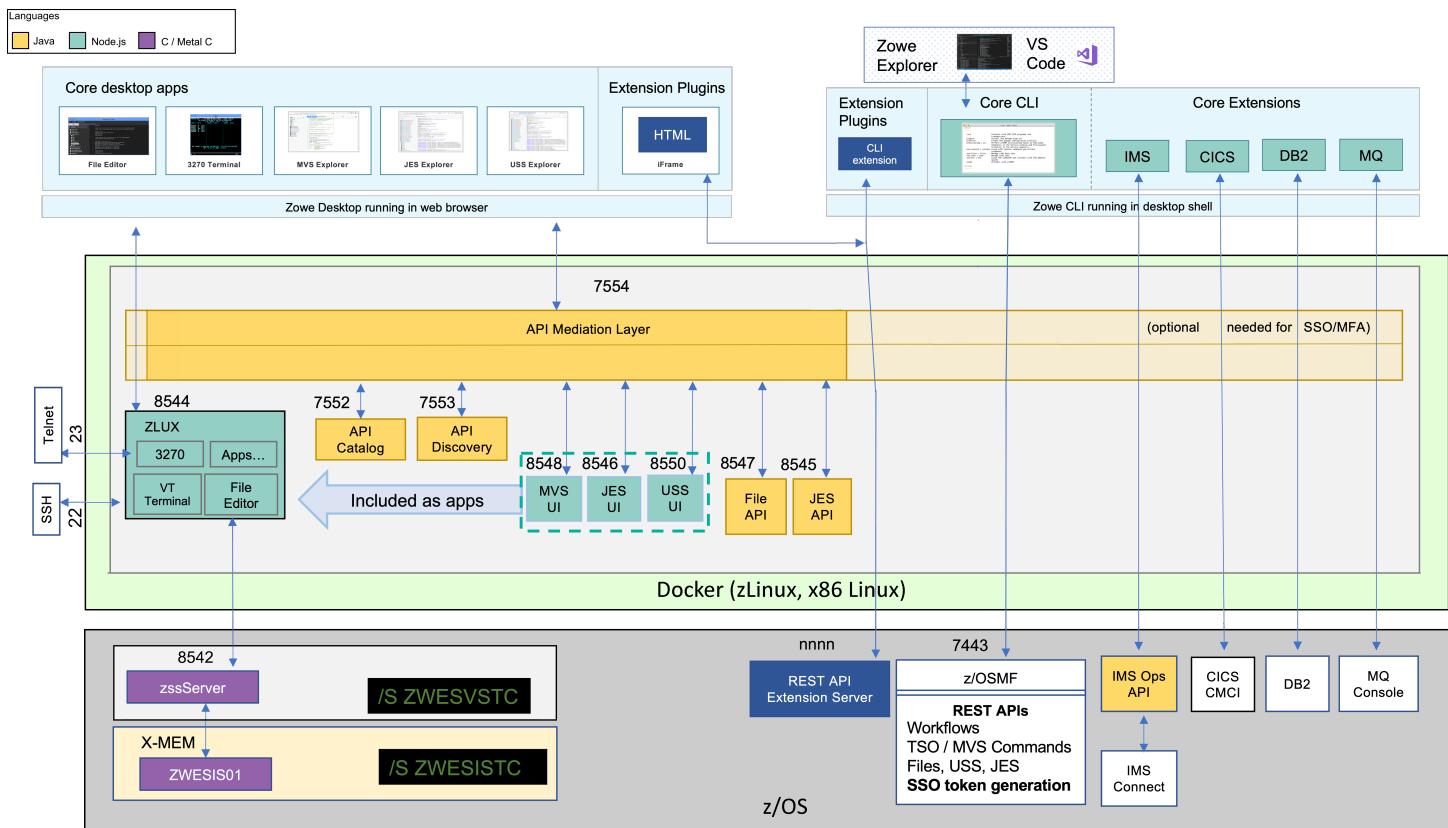
Zowe server components can be installed and run entirely on z/OS, but a subset of the components can alternatively run on Linux or z/Linux via Docker. While on z/OS, many of these components run

under Unix System Services (USS). The ones that do not run under USS must remain on z/OS when using Docker in order to provide connectivity to the mainframe.

Zowe architecture when using Docker image

Technical Preview The Zowe Docker build is a technical preview.

The following diagram depicts the difference in locations of Zowe components when using Docker as opposed to running all components on z/OS.

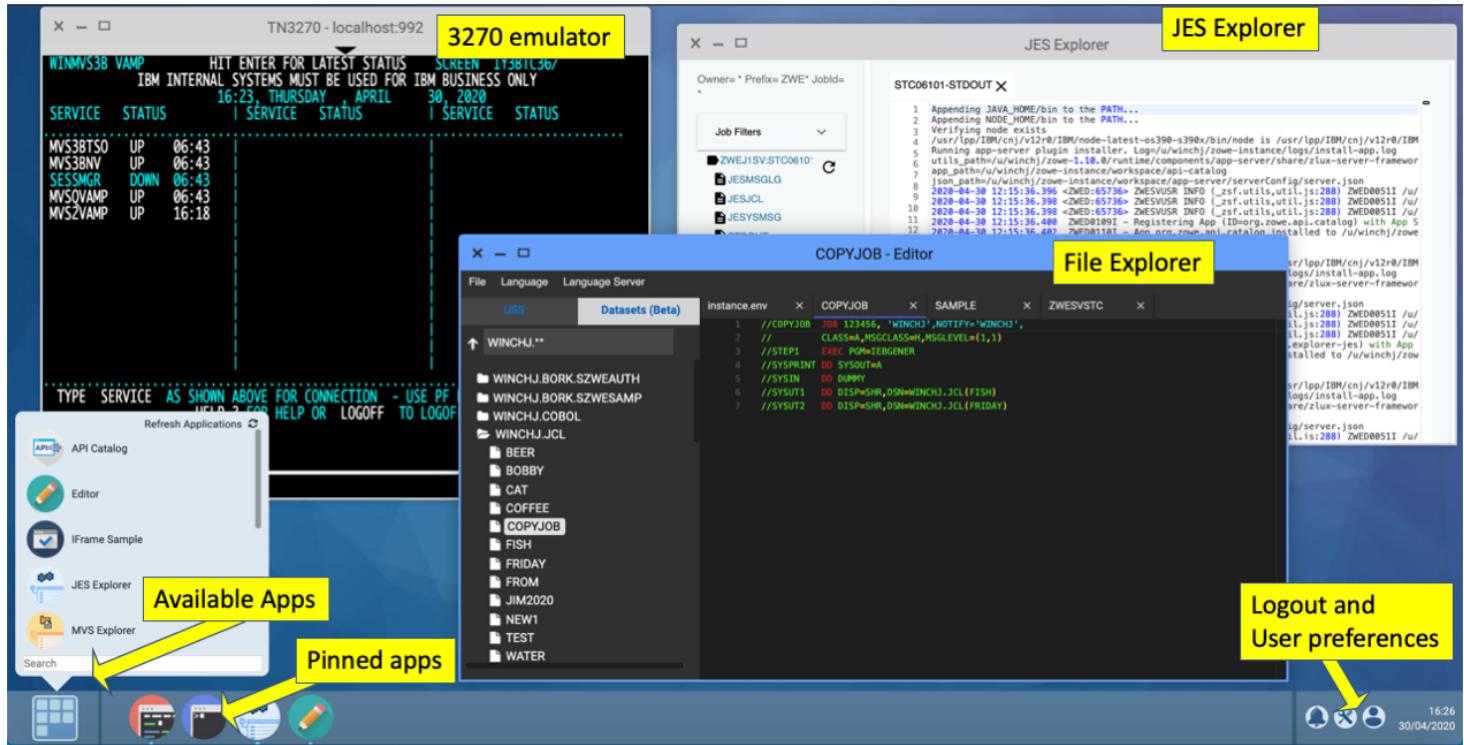


The components on z/OS run under the Zowe started task `ZWESVSTC`, which has its own user ID `ZWESVUSR` and includes a number of servers each with their own address space. The `ZWESVSTC` started task has a `STDOUT` file that includes log and trace information for its servers. Sever error messages are written to `STDERR`. For problem determination, see [Troubleshooting](#).

When Docker is used, server components not running on z/OS instead run in a Linux environment provided via the Docker container technology. The servers run as processes within the container which log to `STDOUT` and `STDERR` of that container, with some components also write to the Zowe instance's log directory.

App Server

The App Server is a node.js server that is responsible for the Zowe Application Framework. It provides the Zowe desktop, which is accessible through a web browser via port 8544. The Zowe desktop includes a number of applications that run inside the Application Framework such as a 3270 emulator and a File Editor.



The App Server server logs are written to `<INSTANCE_DIR>/logs/appServer-yyyy-mm-dd-hh-mm.log`. The Application Framework provides REST APIs for its services that are included on the API catalog tile `Zowe Application Framework` that can be viewed at https://<ZOWE_HOST_IP>:7554/ui/v1/apicatalog/#/tile/ZLUX/zlux.

ZSS

The Zowe desktop delegates a number of its services to the ZSS server which it accesses through the http port 8542. ZSS is written in C and has native calls to z/OS to provide its services. ZSS logs are written `STDOUT` and `STDERR` for capture into job logs, but also as a file into `<INSTANCE_DIR>/logs/zssServer-yyyy-mm-dd-hh-mm.log`.

API Gateway

The API Gateway is a proxy server that routes requests from clients on its northbound edge, such as web browsers or the Zowe command line interface, to servers on its southbound edge that are able to

provide data to serve the request. It is also responsible for generating the authentication token used to provide single sign-on (SSO) functionality. The API Gateway homepage is

https://<ZOWE_HOST_IP>:7554, that after authentication allows you to navigate to the API Catalog.

The screenshot shows the 'API Mediation Layer' homepage. The top navigation bar is blue with the Zowe logo and the text 'API Mediation Layer'. The main content area has a white background with a large title 'API Mediation Layer' in bold black font. Below the title is a bulleted list of three items, each preceded by a green checkmark:

- ✓ The API Catalog is running
- ✓ The Discovery Service is running
- ✓ The Authentication service is running

API Catalog

The API Catalog provides a list of the API services that have registered themselves as catalog tiles. These allow you to view the available APIs from Zowe's southbound servers as well as test REST API calls.



Available API services



API Mediation Layer API

The API Mediation Layer for z/OS internal API services. The API Mediation Layer provides a single point of access to mainframe REST APIs and offers enterprise cloud-like feature...

● All services are running

z/OS Datasets and Unix Files services

IBM z/OS Datasets and Unix Files REST services

z/OS Jobs services

IBM z/OS Jobs REST services

● All services are running

z/OSMF services

IBM z/OS Management Facility REST services

● All services are running

Zowe Application Server

The Proxy Server is an HTTP, HTTPS, and Websocket server built upon NodeJS and ExpressJS. This serves static content via "Plugins", and is extensible by REST and Websocket "Data..."

● All services are running

API Discovery

The API Discovery server acts as the registration service broker between the API Gateway and its southbound servers. It can be accessed through the URL https://<ZOWE_HOST_IP>:7552. You can view a list of registered API services on the API discovery homepage.

System Status

Environment	test	Current time	2020-04-30T14:50:51 +0000
Data center	default	Uptime	02:35
		Lease expiration enabled	true
		Renews threshold	18
		Renews (last min)	44

DS Replicas

winmvs3b.hursley.ibm.com

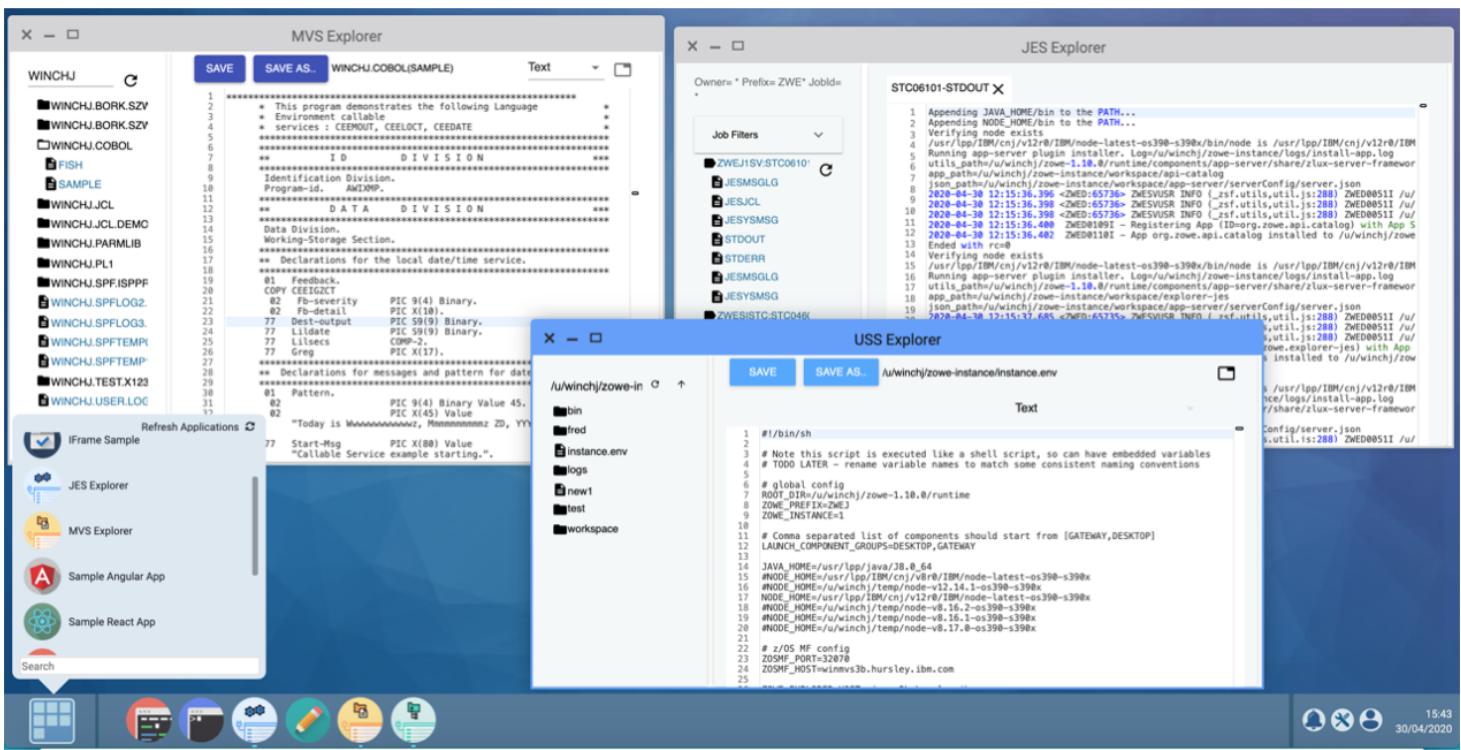
Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
APICATALOG	n/a (1)	(1)	UP (1) - winmvs3b.hursley.ibm.com:apicatalog:26500
DATASETS	n/a (1)	(1)	UP (1) - STATIC- winmvs3b.hursley.ibm.com:datasets:26504
DISCOVERY	n/a (1)	(1)	UP (1) - winmvs3b.hursley.ibm.com:discovery:26501
EXPLORER-JES	n/a (1)	(1)	UP (1) - STATIC- winmvs3b.hursley.ibm.com:explorer-jes:26505
EXPLORER-MVS	n/a (1)	(1)	UP (1) - STATIC- winmvs3b.hursley.ibm.com:explorer-mvs:26506
EXPLORER-USS	n/a (1)	(1)	UP (1) - STATIC- winmvs3b.hursley.ibm.com:explorer-uss:26507
GATEWAY	n/a (1)	(1)	UP (1) - winmvs3b.hursley.ibm.com:gateway:26502
JBS	n/a (1)	(1)	UP (1) - STATIC- winmvs3b.hursley.ibm.com:jobs:26503
UNIXFILES	n/a (1)	(1)	UP (1) - STATIC- winmvs3b.hursley.ibm.com:unixfiles:26504
ZLUX	n/a (1)	(1)	UP (1) - localhost:zlux:26508
ZOSMF	n/a (1)	(1)	UP (1) - STATIC- winmvs3b.hursley.ibm.com:zosmf:32070

General Info

MVS, JES, and USS UI

Zowe provides a number of rich GUI web applications for working with z/OS. This includes the MVS Explorer for data sets, the JES Explorer for jobs, and the USS Explorer for the Unix File System. You can access them through the Zowe desktop.



File API and JES API

The File API server provides a set of REST APIs for working with z/OS data sets and Unix files. These APIs are used by the MVS and USS Explorer apps.

The JES API server provides a set of REST APIs for working with JES. These APIs are used by the JES Explorer application.

Both the File API and JES API servers are registered as tiles on the API catalog, so you can view the Swagger definition and test API requests and responses.

Cross memory server

The Cross memory server is a low-level privileged server for managing mainframe data securely. For security reasons, it is not an HTTP server. Instead, it has a trust relationship with ZSS. Other Zowe components can work through ZSS in order to handle z/OS data that would otherwise be unavailable or insecure to access from higher-level languages and software.

Unlike all of the servers described above which run under the `ZWESVSTC` started task as address spaces for USS processes, the cross memory server has its own separate started task `ZWESISITC` and its own user ID `ZWESIUSR` that runs the program `ZWESIS01`.

Frequently Asked Questions

Check out the following FAQs to learn more about the purpose and function of Zowe™.

- [Zowe FAQ](#)
- [Zowe CLI FAQ](#)
- [Zowe Explorer FAQ](#)

Zowe FAQ

What is Zowe?

 Click to open answer

Who is the target audience for using Zowe?

 Click to open answer

What language is Zowe written in?

 Click to open answer

What is the licensing for Zowe?

 Click to open answer

Why is Zowe licensed using EPL2.0?

 Click to open answer

What are some examples of how Zowe technology might be used by z/OS products and applications?

 Click to open answer

What is the best way to get started with Zowe?

 Click to open answer

What are the prerequisites for Zowe?

 Click to open answer

What's the difference between using Zowe with or without Docker?

Technical Preview

 Click to open answer

Is the Zowe CLI packaged within the Zowe Docker download?

Technical Preview

 Click to open answer

Does ZOWE support z/OS ZIIP processors?

 Click to open answer

How is access security managed on z/OS?

 Click to open answer

How is access to the Zowe open source managed?

 Click to open answer

How do I get involved in the open source development?

 Click to open answer

When will Zowe be completed?

 Click to open answer

Can I try Zowe without a z/OS instance?

 Click to open answer

Zowe CLI FAQ

Why might I use Zowe CLI versus a traditional ISPF interface to perform mainframe tasks?

 Click to open answer

With what tools is Zowe CLI compatible?

 Click to open answer

Where can I use the CLI?

 Click to open answer

Which method should I use to install Zowe CLI?

 Click to open answer

How can I get help with using Zowe CLI?

 Click to open answer

How can I use Zowe CLI to automate mainframe actions?

 Click to open answer

How can I contribute to Zowe CLI?

 Click to open answer

Zowe Explorer FAQ

Why might I use Zowe Explorer versus a traditional ISPF interface to perform mainframe tasks?

 Click to open answer

How can I get started with Zowe Explorer?

 Click to open answer

Where can I use Zowe Explorer?

 Click to open answer

How do I get help with using Zowe Explorer?

 Click to open answer

How can I use Secure Credential Store with Zowe Explorer?

 Click to open answer

How can I use FTP as my back-end service for Zowe Explorer?

 Click to open answer

How can I contribute to Zowe Explorer?

 Click to open answer

Release notes

Learn about what is new, changed, or removed in Zowe™.

Zowe Version 1.21 and earlier releases include the following enhancements, release by release.

- [Version 1.21.0 LTS \(April 2021\)](#)
- [Version 1.20.1 LTS \(March 2021\)](#)
- [Version 1.20.0 LTS \(March 2021\)](#)
- [Version 1.19.1 LTS \(February 2021\)](#)
- [Version 1.19.0 LTS \(February 2021\)](#)
- [Version 1.18.0 LTS \(January 2021\)](#)
- [Version 1.17.0 LTS \(November 2020\)](#)
- [Version 1.16.0 LTS \(October 2020\)](#)
- [Version 1.15.0 LTS \(September 2020\)](#)
- [Version 1.14.0 LTS \(August 2020\)](#)
- [Version 1.13.0 LTS \(July 2020\)](#)
- [Version 1.12.0 LTS \(June 2020\)](#)
- [Version 1.11.0 LTS \(May 2020\)](#)
- [Version 1.10.0 LTS \(April 2020\)](#)
- [Version 1.9.0 LTS \(February 2020\)](#)
- [Version 1.8.1 \(February 2020\)](#)
- [Version 1.8.0 \(February 2020\)](#)
- [Version 1.7.1 \(December 2019\)](#)
- [Version 1.7.0 \(November 2019\)](#)
- [Version 1.6.0 \(October 2019\)](#)
- [Version 1.5.0 \(September 2019\)](#)
- [Zowe SMP/E Alpha \(August 2019\)](#)
- [Version 1.4.0 \(August 2019\)](#)
- [Version 1.3.0 \(June 2019\)](#)
- [Version 1.2.0 \(May 2019\)](#)
- [Version 1.1.0 \(April 2019\)](#)

- Version 1.0.1 (March 2019)
- Version 1.0.0 (February 2019)

Version 1.21.0 LTS (April 2021)

New features and enhancements

Zowe installation and configuration

- Introduced a new non-strict verify certificates mode which can be customized as `NONSTRICT_VERIFY_CERTIFICATES` in `zowe-setup-certificates.env`. Comparing to strict `VERIFY_CERTIFICATES` mode, this non-strict mode will not validate certificate Common Name or Subject Alternative Name (SAN). However, Zowe will still validate if the certificate authorities are trusted in the trust store. This change was introduced with issue [zowe/api-layer#1334](#) and fixed by [#2062](#).
- Added two new JCLs (`ZWESSKTK` and `ZWENOSSO`) to the PDS sample library `SZWESAMP`. `ZWESSKTK` can be used to create SSO `PKCS#11` token and set up required security configurations. `ZWENOSSO` can be used to remove the `PKCS#11` token and related security changes. This issue is described in [zowe-install-packaging#2052](#) and fixed by [#2094](#).
- Reduced the amount of checking of Java and node levels. [#1997](#), [#2063](#)
- Added a list to the ZWESECUR JCL for the client cert PERMIT to match the other RACF commands. [#1971](#), [#2063](#)

Zowe API Mediation Layer

- The dockered deployment of Zowe now supports Redis as an off-platform storage for the Caching service. ([a7f4ad](#)), [#1128](#)
- Configuration of the API ML run is now permitted where the hostname in the certificate is not verified in a strict manner. The certificate Common Name or Subject Alternate Name (SAN) are not checked. This facilitates deployment to Marist when certificates are valid but do not contain a valid hostname. This configuration is for development purposes only and should not be used for production. ([2da761a](#)), [#1334](#)
- Caching service: The alphanumeric constraint for keys stored in the service has been removed. ([237420](#))[#1317](#)
- An endpoint has been added to delete all keys for a specific service. ([0c3e01](#)), [#1253](#)

Zowe App Server

- app-server now supports reaching ZSS through TLS without the need for AT-TLS, by reading new properties within the "agent" config [#151](#)
- Synchronize user preference setting for TLS verification so that app-server matches the value used by APIML; VERIFY_CERTIFICATES [#165](#)
- ZSS cookie is now sent to the browser, rather than having the app-server mediate the ZSS connection, so that ZSS can be used through APIML in the case that SSO is not enabled. [#274](#)
- ZSS now uses HTTPS by default, rather than relying upon AT-TLS for the purpose. AT-TLS is still available, but now unless ZOWE_ZSS_SERVER_TLS=false is set, ZSS will use the keystore data for its HTTPS configuration, and when HTTPS is enabled will bind to ZOWE_EXPLORER_HOST value, as the other servers do, instead of 127.0.0.1 by default as ZSS would for HTTP.
- Added function to toggle the file explorer
- Added calls to the function in open file and open dataset so that when the user inputs the string with a true in the url it will hide the file explorer
- Added a global "environment" object in Zowe ZLUX which allows for retrieving select environment properties from the zowe instance for conditional decision-making
- The Zowe desktop uses the new environment object to determine whether to contact ZSS through app-server or through APIML depending on if ZSS is found on APIML
- app-server will contact ZSS through APIML if APIML is enabled and app-server finds that APIML is accessible from APIML
- sso-auth plugin no longer keeps ZSS cookie within app-server; the cookie will now be sent to and used from the browser to facilitate high availability

Zowe CLI

The following enhancements were added to the **core CLI**:

- Added the option `--jcl-symbols` to the jobs submit command to enable users to specify JCL symbol names and values.
- Made changes to definition files for zowe ssh commands [#603](#)
- Added a standard data set template with no parameters set.

The following enhancement was added to the **Imperative CLI Framework**:

- Added `headers[]` option to `TextUtils.getTable()`. [#369](#)

Zowe Explorer

- Added the Issue TSO Commands feature [#1245](#)

Bug fixes

Zowe installation and configuration

- Fixed an [issue](#) where Zowe runs with the prefer IP address parameter set true as default. [#2063](#).
- Important!** With this fix, the `APIML_PREFER_IP_ADDRESS` configuration in `instance.env` is deprecated and Zowe will always use `false` as the value.
- Fixed an [issue](#) that the Zowe 1.19.1 `zowe-setup-certificates.sh` script failed if not executed from the correct dir. [#2062](#)
- Fixed script `zowe-support.sh` not working. [#2049](#)
- Fixed `validate_certificate_domain` reports false negative on wildcard domains. [#2117](#)

Zowe API Mediation Layer

- Stop leaking X-Certificate headers ([b2737a](#))[\[#1328\]](#)[\(<https://github.com/zowe/api-layer/pull/1328>\)](https://github.com/zowe/api-layer/pull/1328)
- Remove the wait from start.sh to reduce address spaces ([2ba780](#))[#1335](#)
- Make the version endpoint available at the URL: /application/version ([0ac95a4](#))[#1312](#)
- Load the JWT secret properly when concurrently loaded and requested ([1644a8c](#), [#1255](#))
- Swagger v2 yaml parsed and rendered ([a1f2cc0](#)) [#1229](#)

Zowe App Server

- app-server now registers to APIML using the fully qualified hostname found from `ZOWE_EXPLORER_HOST` or `ZOWE_EXTERNAL_HOSTS`. Previously, it used the short hostname reported by the OS. This resolves bugs about hostname mismatch. [#153](#)
- Set the hostname used for Eureka to match the value of `ZWE_EXTERNAL_HOSTS` if exists, or otherwise `ZOWE_EXLORER_HOST`, for the purpose of avoiding certificate verification issues between app-server and APIML under certain circumstances
- Set the cookie path to root in order to avoid multiple cookies when the browser tries to set path automatically
- Use the hostname given by Zowe config in order to avoid errors from the hostname certificate matching when accessing the app server through APIML
- ZSS tile on the API Catalog has been fixed due to HTTPS mode listening on the hostname of `ZOWE_EXPLORER_HOST` to align with the other zowe servers.

- ZSS dataset write REST API has improved handling of when a client sends fixed dataset content with too little padding [#209](#)

Zowe CLI

The following bugs were fixed in the **Imperative CLI Framework**:

- Print a subset of the `stdout` and `stderr` buffers when calling `mProgressApi`'s `endBar()` to prevent duplication of output.
- Replaced `this` with `ImperativeConfig.instance` in `ImperativeConfig.getCallerFile()`. [#5](#)

The following bugs were fixed in the **Secure Credential Store Plug-in**:

- Updated the Keytar and prebuild-install dependencies to make offline installation possible for npm@7 users.

The following bugs were fixed in the **FTP Plug-in**:

- Fixed list jobs problems.
- Updated list jobs unit test and system test.

Zowe Explorer

- Fixed the issue that caused the USS tree to collapse after renaming a folder [#1259](#)
- Fixed the issue that prevented jobs with an octothorpe (#) in the name from opening [#1253](#)

Version 1.20.1 LTS (March 2021)

Bug fixes

Zowe Installation and Configuration

- Fixed an issue when importing external certificate authorities. [#2032](#)

Version 1.20.0 LTS (March 2021)

New features and enhancements

Zowe API Mediation Layer

- Feature: x509 authentication scheme. This feature supports authentication with a client certificate in southbound services whereby users can decide which part of the certificate to use. (#1208) ([94dbf37](#)), closes [#1208](#)
- Feature: Add NodeJS sample service and enabler. This feature makes it possible for a service based on NodeJS to register with API ML in a similar way as with other onboarding enablers. (#1140) ([c86a289](#)), closes [#1140](#)
- Feature: Allow Zowe to run without a jwtsecret if the jwtsecret is not required. (#1203) ([7dc6dad](#)), closes [#1203](#)
- Feature: Introduce token validation providers. This feature provides a future mechanism of token validation whereby custom endpoint can be provided that requires authentication. (#1142) ([80cc790](#)), closes [#1142](#)
- Feature: Reject eviction strategy has been added to VSAM. If storage is full, and `ZWE_CACHING_EVICTION_STRATEGY` is set to `reject` this feature prevents the Caching Service from removing entries, but returns a status code 507 with the message, "Insufficient storage space limit". (#1112) ([70c2d71](#)), closes [#1112](#) ([80cc790](#)), closes [#1142](#)
- Feature: Base information (SSO, API ID) about a service is now displayed in the API Catalog. (#1116) ([4b61377](#)), closes [#1116](#) [#1116](#)
- Feature (Caching Service): Production logging for the Caching Service. This feature limits messages sent to Spool to the bare minimum, thereby improving the information returned to the caller. (#1185) ([7adfffb1](#)), closes [#1185](#)

Zowe App Server

- Added a manifest file, a validate script, and refactored configure, start, and app-server scripts to better conform to Zowe lifecycle management standards [#144](#)
- Added a method to read user and group server timeout information from a JSON file [#248](#)
- Added a quick search to the File Tree that filters opened files, folders, and datasets [#197](#)
- Added a preferences menu to customize the editing behavior and color theme. The preferences can be previewed in realtime, but can also be saved to the app-server so that they are applied every time the editor is opened. [#195](#)
- Added undo and redo menu items when editing a file [#195](#)

Zowe CLI

The following enhancements were added to the **core CLI**:

- Updated the Imperative version to handle GZIP compression on REST requests.

- Added a `like` option to the `zowe zos-files create data-set` command. Use this option to like datasets. [#771](#)
- Added a `--protocol` option to allow you to specify the HTTP or HTTPS protocol used. The default value remains HTTPS. [#498](#)
- Added an example for running a Db2 command with the `zowe zos-console issue command` command. [#641](#)

The following enhancement was added to the **Imperative CLI Framework**:

- Added decompression support for REST responses with Content-Encoding `gzip`, `deflate`, or `br`. [#318](#)

The following enhancements were added to the **IBM Db2 Plug-in**:

- Added a semicolon after each sql statement when exporting a table.
- Added a help example for how to pass output values when calling a Db2 stored procedure.

The following enhancement was added to the **FTP Plug-in**:

- Added allocate command to allocate sequential or partitioned dataset.

Zowe Explorer

- Added the monorepo landing Readme that contains the high-level overview of the repository folders such as `packages` folder, instructions on how to contribute to the project and links to Medium articles providing additional useful information about Zowe Explorer and Zowe [#1199](#).
- Added the previously selected `RejectUnauthorized` value to the placeholder text of the entry field while updating an existing profile. In addition, the value is highlighted and shown at the top of the selection list [#1218](#).
- Added the pre-filled and pre-selected filename of the copied member to the entry field while performing the paste member action [#1183](#).
- Added the multiple deletion of jobs feature [#1128](#).
- Improved error handling for the data set copy/paste member, migrate, and recall functions [#1219](#).

For more information about the Zowe Explorer release notes, see [Changelog](#).

Bug Fixes

Zowe API Mediation Layer

- Bugfix (authentication): Support specific z/OSMF version. This fix allows the user to force the authentication token that is used. (#1241) ([2da761a](#)), closes [#1241](#)
- Bugfix (authentication): Ignore wrong or non-existing SAF classes when SAF is not used (#1216) ([c5ea311](#)), closes [#1216](#)
- Bugfix (enabler): Add unregistration method to the the Node.js enabler. (#1214) ([1ecd5c7](#)), closes [#1214](#)
- Bugfix: Enable /api/v1/gateway path format for the `/auth/logout`, `/auth/login`, `/auth/query`, and `/auth/passticket` endpoints (#1126) ([13ac9a5](#)), closes [#1126](#)
- Bugfix: Accept `swagger/openapi` in yaml format (#1202) ([0c412b0](#)), closes [#1202](#)

Zowe CLI

The following bug was fixed in the **core CLI**:

- Removed the conflicting alias `-o` for `--protocol` option.

The following bugs were fixed in the **Imperative CLI Framework**:

- Added `Protocol` to the Error Details coming from the `AbstractRestClient`. (#539)
- Fixed vulnerabilities by replacing marked with markdown-it and sanitize-html.
- Fixed plugin install failing to install package from private registry.

The following bugs were fixed in the the **Secure Credential Store Plug-in**:

- Updated the Keytar dependency to v7 to be compatible with Node.js v15.
- Provided additional instruction in readme for npm@7 users.

Zowe Explorer

- Fixed the issue that prevented the list of recently opened files from being displayed upon request. You can access a list of recently opened files by pressing the Ctrl+Alt+R (Windows) or Command+Option+R (Mac) key combination [#1208](#).
- Fixed the issue that prevented file picker from functioning. The file picker feature lets you filter your datasets in the tree by pressing the Ctrl+Alt+P (Windows) or Command+Option+P (Mac) key combination [#992](#).
- Fixed the issue that caused the content from a previously filtered USS directory instead of the currently filtered USS directory to be served [#1134](#).

For more information about the Zowe Explorer release notes, see [Changelog](#).

Version 1.19.1 LTS (February 2021)

Notable changes

SMPE PTF

With the continuous optimization of the Zowe build size, now the Zowe release can fit in one PTF. This single PTF UO01969 will supersede PTF UO01967 and UO01968 from the v1.19.0 release.

New features and enhancements

Zowe CLI

The following enhancements were added to the **core CLI**:

- Updated Imperative version to support npm@7. This fixes an error when installing plugins.

The following enhancements were added in the **Imperative CLI Framework**:

- Fixed plugin install commands which were broken in npm@7. [#457](#)

Bug fixes

SMPE PTF

- Fixed the missing HOLD data about extra steps the system programmer should take after applying the PTF. It explains new configuration options that are introduced in `instance.env` and also a reconfiguration step where `zowe-configure-instance.sh` is mandatory.

Zowe CLI

The following bugs were fixed in the **Imperative CLI Framework**:

- Fixed incorrect formatting of code blocks in web help. [#535](#)

Version 1.19.0 LTS (February 2021)

Notable changes

Package manifest and component installer

Each Zowe extension and each core component can now use a manifest file to describe itself. The manifest file defines the name and purpose of the component. It also provides information about how this component should be installed, configured, started and tested. For more information, see [Packaging z/OS extensions](#).

Two tools, `zowe-install-component.sh` and `zowe-configure-component.sh`, are introduced in this release for technical preview. The `zowe-install-component.sh` helps you install any Zowe server component (extension). Zowe core components are also installed with this tool. The `zowe-configure-component.sh` tool helps you configure an installed Zowe server component (extension) for a Zowe instance. Zowe core components are also configured with this tool. In order to be compatible with the tools, it is recommended that the components follow [Zowe server component package format standard](#).

X.509 client certificate authentication support for API Mediation Layer (Technical Preview)

This feature is released for technical preview in Zowe 1.19. Previously, users were required to provide credentials (usually basic authentication) to make a login call against the API Gateway. From release 1.19 you can now use the x509 client certificate for calls to authenticate in the API ML, whereby information from the certificate verifies the user's identity through SAF and then returns a proper JWT.

If you would like to offer feedback about using client certificate authentication, please create an issue against the Zowe `api-layer` repository.

Standalone run of Zowe API Mediation Layer

You can now start the API Mediation Layer independently of other Zowe components. This allows you to minimize the resources used when you use Zowe as a Devops tool instead of a Virtual Desktop tool.

New features and enhancements

Zowe API Mediation Layer

- The connection limit of the Gateway has been configured to support multiple long-running requests by service. [#843](#)
- The size of API Mediation Layer has been reduced to fit within 150MB. [#909](#)
- You can now configure whether or not the Catalog appears on the Gateway homepage [#727](#)
- Connection limits have been enhanced to improve latency times when making requests through the API ML. This feature also enables concurrent requests. [#987](#)

- The connection limit log messages have been enhanced. New messages indicate when too many connections occur. [#987](#)
- The `/api/v1/gateway/services/{serviceId}` endpoint has been added which provides information about a service in API ML for API clients. You can now view information to choose the applicable available API service without having a trusted service certificate. Proper SAF authorization is required. [#873](#)
- The size limitation in the InMemory cache for proper handling is now supported when size limitations are reached. [#998](#)
- The 'Remove Oldest' eviction mechanism for Caching Service has been implemented to limit the volume of data in the cache. [#998](#)
- CORS origins per service has been configured so that onboarded services can request that CORS behavior for a route be delegated to the API Mediation Layer. [#997](#)
- The 'Reject eviction' strategy to the Caching Service has been implemented to limit the volume of data in the cache. [#998](#)
- Debug logging to x.509 Client certificate authentication classes has been added. This feature enables users to determine the cause of system problems during client certificate authentication setup.

Zowe App Server

- Dispatcher actions have been added to the iFrame adapter. [#302](#)
- Support has been added to a new destination property for iFrame pluginDefinition.json. iFrame with the new destination property will now make requests to /web/iFrame. [#296](#)
- The compression-webpack-plugin has been updated from 3.1.0 to 4.0.0 [#304](#)
- Support for a new destination property to iFrame pluginDefinition.json has been added, as well as a new double iFrame default template. [#257](#)
- Axios has been updated from 0.19.2 to 0.21.1 in `/test/webapp/websocket` [#259](#)
- The option to refresh file content has been added to the Editor. [#185](#)
- Refresh buttons have been added to USS and MVS. [#108](#)
- Additional keybindings and other improvements have been added to the Editor. [#182](#)
 - You can now move between open file tabs by using the following hotkeys: **Alt + (PAGEUP or <)** and **Alt + (PAGEDOWN or >)**
 - After closing a tab, or multiple tabs, you can now undo the close by using the following hotkeys: **CTRL + ALT + T**
 - The search function hotkey has been changed from **ALT + S** to **ALT + S.**
 - You can now hide/show the File Tree by using the following hotkeys: **ALT + B.**

- Existing code highlighters have been reorganized in order to improve their readability. Additionally, a new code highlighter for the REXX language has been added. This new code highlighter detects files and datasets wherein the files should end with the .rex prefix, but the datasets may contain the rex or exec qualifiers. [#181](#)

Zowe Explorer

- Updated Keytar and Jest dev deps for Node 14.

Zowe JES/MVS/USS Explorers

The following features and enhancements were added to the **JES Explorer**:

- Introduced the menu shortcuts and confirmation dialog before canceling or purging the job for JES explorer. [#235](#)
- Refactored JES packaging and installation scripts, and folder renames to accommodate new iframe capability in ZLUX. [#236](#)
- Added manifest for API ML and App Framework installation using new plugin installation process. [#234](#)

The following features and enhancements were added to the **MVS Explorer**:

- Refactored MVS packaging and installation scripts, and folder renames, to accommodate new iframe capability in ZLUX. [#164](#)
- Added manifest for API ML and App Framework installation using new plugin installation process. [#164](#)

Bug Fixes

Zowe API Mediation Layer

- API ID is not sent to Eureka in metadata by the Java enabler [#991](#)
- Fixed tcp connections that are stuck open. [#1009](#)

Zowe App Server

- In previous versions, sso-auth URL encoding that used the % sign would always return with authorization:false when using RACF. This issue has been resolved in this version. [#258](#) [#27](#)
- Fixes a bug in the Editor that prevented the unsaved changes symbol from being displayed. [#185](#)
- Stopped event propagation in the Editor in order to resolve a conflict with Firefox. [#183](#)

- Fixes a bug in the Editor that would cause the Languages menu to disappear when closing all tabs, then clicking undo. [#182](#)

Zowe CLI

The following bug was fixed in the **core CLI**:

- Updated the Imperative version to fix a vulnerability.

The following bugs were fixed in the **Imperative CLI Framework**:

- Fixed vulnerabilities by updating `marked` [#515](#)
- Fixed an issue where `TypeError` has been raised by `Logger.getCallerFileAndLineTag()` when there was not filename for a stack frame. [#449](#)

Version 1.18.0 LTS (January 2021)

Notable changes

Zowe Docker build Technical Preview is available

The Zowe Docker build enables you to run a subset of the Zowe server-side components outside z/OS. The Docker build runs in combination with the convenience or SMP/E build. You can download the build directly via the .tar file, or as a cloud download from Docker Hub. Separate downloads exist for z/Linux ("s390x") and other Linux ("amd64" for intel & amd systems).

For more information, see [Docker Installation Roadmap](#). To download the Docker build technical preview, visit [Zowe.org](#).

New features and enhancements

Zowe API Mediation Layer

- Version 1.18.0 introduces a feature allowing users to run the Zowe API Mediation Layer as a standalone component. After downloading and installing the current Zowe SMPE package, you can then configure and deploy only the Zowe API Mediation Layer without the other Zowe components. [#856](#)
- You can now configure more detailed logging outside of Spool. [#709](#)

- High Availability: The start script per API ML service has been componentized. You can now launch and restart API Mediation Layer components individually. [#862](#)
- High Availability: It is now possible to distinguish between internal and external traffic through port separation, whereby each port uses a unique certificate; one presenting an internal certificate, and the other an external certificate. [#910](#)
- API version is now automatically set to the version tab selected in the API Catalog so users can easily grab the Base Path. [#943](#)
- API Catalog versioning has been improved with the addition of the API differences tab. This feature enables you to compare versions of two APIs. [#923](#)

Zowe App Server

- The Zowe App Framework's "single app mode" is now based on code shared with the Desktop, allowing it to support the Desktop's notification API and app2app communication. [#67](#) [#292](#)
 - This is backward compatible with apps that have previously used single app mode.
 - In the case where app2app communication is used and spawns a second app, that app will spawn in a window but will not be able to be minimized due to single app mode having no Desktop, and therefore no way to restore a minimized window.
- ZSS plug-ins can now issue HTTP requests as HTTP clients, provided by a new library in zowe-common-c. [#179](#)

Zowe CLI

The following enhancements were added to the **core CLI**:

- Added a `--replace` option to the `zowe zos-files copy data-set` command. Use this option if you want to replace like-named members in the target data set. [#808](#)
- Improved a cryptic error message that was shown if the TSO address space failed to start for the `zowe zos-tso issue command` command. [#28](#)

The following enhancements were added to the **Imperative CLI Framework**:

- Added an `arrayAllowDuplicate` option to the `ICommandOptionDefinition` interface. By default, the option value is set to `true` and duplicate values are allowed in an array. Specify `false` if you want Imperative to throw an error for duplicate array values. [#437](#)
- Expose `trim` parameter from `wrap-ansi` within `TextUtils.wordWrap()`

The following enhancement was added to the **IBM Db2 Plug-in**:

- Added a help example for how to pass output values when calling a Db2 stored procedure.

The following enhancement was added to the **FTP Plug-in**:

- Move the reusable code from handlers to api folder.

Zowe JES/MVS/USS Explorers

The following features and enhancements were added to the **JES Explorer**:

- Added webdevSever proxy setting in webpack.config.js to enable https for local development.

Bug Fixes

Zowe API Mediation Layer

- ZaasJwtService enhancement on JWT parsing and error handling. [#897](#)
- Upgrade dependencies for the Enablers. [#933](#)

Zowe App Server

- The zss server log verbosity seen when using the TN3270 desktop app has been reduced. [#188](#)
- Keep-alive parsing has been temporarily disabled to patch a memory leak. A permanent fix that will allow the use of keep-alive parsing is scheduled to be implemented in the next release. [#186](#)
- The warning messages on the zss server startup have been removed due to a shell syntax problem. [#238](#)
- In previous versions, static app2app recognizers would not be loaded from storage because they were treated as actions instead of as recognizers. This issue has been resolved in this release. [#297](#)

Zowe CLI

The following bugs were fixed in the **core CLI**:

- Removed "[object Object]" text that appeared in some error messages. The proper text "Imperative API Error" is now displayed. [#836](#)
- Improved performance of `zowe zos-files list` commands when long lists are printed to console. [#861](#)
- Updated Imperative dependency version to one that does not contain a vulnerable dependency.

The following bug was fixed in the **Imperative CLI Framework**:

- Updated `opener` dependency due to command injection vulnerability on Windows - [GHSL-2020-145](#)

Version 1.17.0 LTS (November 2020)

Notable changes

z/OSMF workflow for configuring Cross Memory Server

You can now use the z/OSMF workflow to install, configure, and launch the cross memory server if you want to use the Zowe desktop. The z/OSMF workflow also lets you create APF-authorized load libraries that are required to install and configure the cross memory server. For more information, see [Configure Zowe Cross Memory Server with z/OSMF workflow](#).

Zowe Client SDKs

A new Zowe incubation project - the Zowe Client SDKs (Software Development Kits) is now available for Node.js, Python, and Swift programming languages. You can leverage these SDKs to rapidly develop off-platform applications and automation.

For more information about the Node.js and Python SDKs, see [Using Zowe SDKs](#). For more information about the Swift SDK, see the [Swift SDK Readme](#).

New features and enhancements

The following features and enhancements were added.

Zowe installation

- You can now start ZSS independent of the Zowe Application Framework server by specifying the `LAUNCH_COMPONENT_GROUP "ZSS"`. If `DESKTOP` is specified instead of `ZSS`, ZSS will still be included as a prerequisite to the Application Framework server. [#1632](#)
- Zowe instance configuration script (`zowe-configure-instance.sh`) can now skip checking for Node.js by passing in the `-s` flag since Node.js may not be needed if the components to be launched don't require it. [#1677](#)
- The `run-zowe.sh` script can also skip the checking for Node.js by setting the environment variable `SKIP_NODE=1` for the cases where the components to be launched don't require Node.js.
- Exported the `EXTERNAL_CERTIFICATE_AUTHORITIES` variable to the `zowe-certificates.env` file such that it may be used by the Application Framework server. [#1742](#)

- A new documentation chapter [Upgrading the z/OS System for Zowe](#) has been included, that describes the steps to take when upgrading an existing Zowe installation.

Zowe API Mediation Layer

- Multiple versions of one API are now presented in the Catalog if configured to do so. Users can now switch between different versions within the Catalog to see differences in API documentation between versions. [#844](#)
- Setting `APIML_DEBUG_MODE_ENABLED` in `instance.env` is properly passed on to the all API ML services. [#901](#)

Zowe App Server

- ZSS no longer requires NodeJS for its configure.sh script.
- Added support for DER-encoded X.509 certificates.
- You are now able to change tags for all files in the directory excluding subdirectories. For example, `POST /unixfile/ctag/u/user/tmp?codeset=1047&type=text&recursive=false` should change tags only for files in `u/user/tmp` without changing tags for files in subdirectories. [#176](#)
- Multiple enhancements in the Editor for USS file and directory actions, including: [#84](#) [#102](#) [#93](#)
 - The ability to cut, copy, & paste files into a directory, such as the currently active directory.
 - Re-ordered context menu options.
 - Improved error messages by including more detail.
 - Added support to see a file's current tag and change it.
 - chown & chmod enhancement that pre-populates the owner and group fields when opening the ownership and properties dialogs. It also adds owner and group information to the file properties dialog.

Zowe CLI

The following enhancements were added to the **core CLI**:

- Zowe CLI was tested and confirmed to be compatible with Node.js v14.
- Published the programmatic interfaces in Zowe CLI as separate Software Development Kits (SDKs). [#750](#)
- The "@zowe/cli" package still includes both API and CLI methods. In addition, the following SDK packages are now available:

- @zowe/provisioning-for-zowe-sdk
- @zowe/zos-console-for-zowe-sdk
- @zowe/zos-files-for-zowe-sdk
- @zowe/zos-jobs-for-zowe-sdk
- @zowe/zos-tso-for-zowe-sdk
- @zowe/zos-uss-for-zowe-sdk
- @zowe/zos-workflows-for-zowe-sdk
- @zowe/zosmf-for-zowe-sdk
- @zowe/core-for-zowe-sdk

The following enhancement was added to the **Imperative CLI Framework**:

- Exposed the `trim` parameter from `wrap-ansi` within `TextUtils.wordWrap()`. [#458](#)

The following enhancement was made to enable support for Node.js v14 for the **Secure Credential Store Plug-in**:

- Enabled support on Node.js v14 by updating Keytar dependency to v6. [#28](#)

The following enhancement was made to enable support for Node.js v14 for the **IBM Db2 Plug-in**:

- Enabled support for Node.js v14 by updating dependencies. [#60](#)

Zowe Explorer

- Added login and logout functions for base profiles. You can now log in to API Mediation Layer and generate a token for your base profile. [#914](#)

Zowe JES/MVS/USS Explorers

The following features and enhancements were added to the **JES Explorer**:

- Added ability to refresh content of an open job output file via context menu entry on the job file [#549](#)
- Major material ui update from v1.x to 4.x, and minor react update. Accordion and Snackbar changes as required by latest material-ui version.

Bug fixes

The following bugs were fixed.

Zowe API Mediation Layer

- Improved returned information while logging out via logout on Gateway. [#831](#)
- Updated API paths for the API ML in the API Catalog to use the service id in front. [#853](#)

Zowe App Server

- Make use of external certificate authorities referenced during keystore setup time.
- ZSS startup would issue warnings about failure to write yml files for APIML in the case APIML was not also being used.
- Bugfix: In previous versions, external certificate authorities were not registered with the app server properly and would sometimes contribute to a SELF_SIGNED_CERT_IN_CHAIN error when using the mediation layer. This issue has been resolved by adding external CA certs to the app-server CA array. [#138](#)

Zowe CLI

The following bug was fixed in the **core CLI**:

- Fixed incorrect syntax of example for `zowe files create data-set-vsam` in the help. [#823](#)

The following bug was fixed in the **Imperative CLI Framework**:

- Updated `opener` dependency due to command injection vulnerability on Windows. For more information, see [GHSL-2020-145](#).

Zowe Explorer

- Fixed the empty profile folders in Favorites issue. [#1026](#)
- Fixed the initialization error that occurred when base profiles were used while being logged out from API ML. [#1063](#)
- Fixed the issue preventing the tree refresh function from updating extender profiles. [#1078](#)

Version 1.16.0 LTS (October 2020)

Notable changes

Certificate management and keyring support

- In V1.15, the JCL member `ZWEKRING` was added to the sample PDS library `SZWESAMP`. This member contains commands to create a keyring that can contain the Zowe certificate(s) and a local certificate authority. In this release, the JCL member `ZWENOKYR` was added to `SZWESAMP` that contains the inverse commands, so it can be used to remove the keyring, the Zowe certificate(s), and the certificate authority.
- In V1.15, the JCL member `ZWEKRING` and the supporting code in the Zowe runtimes for working with certificates held in keyrings were provided in beta format for early technical preview for RACF only. In this release, the commands in `ZWEKRING`, `ZWENOKYR` and the supporting code in the Zowe runtimes for working with keyrings and certificates in RACF, TopSecret, and ACF/2 are now a supported piece of functionality.
- A new documentation section is added to help you understand the configuration scenarios around Zowe certificates, and the relationship to a Zowe instance directory and Zowe runtime. See [Topology for the Zowe z/OS launch process](#).

Additional TN3270 terminal configuration options

Additional TN3270 terminal configuration options can now be specified within the `instance.env` configuration file. These choices, such as codepage and terminal dimensions, affect server defaults but do not change the pre-existing ability for you to set your own preferences within the Desktop at runtime. A list of the available options can be found [here](#).

New features and enhancements

The following features and enhancements were added.

Zowe installation

- Moved explorer-ui-server out of explorers into new `shared` folder under Zowe Runtime Directory. [#1545](#), [#207](#), [#37](#)
- Created `zowe-setup-keyring-certificates.env` and removed the overloaded properties from `zowe-setup-certificates.env` to try to simplify the user experience when setting up certificates in the keyring and USS keystore modes. [#1603](#)

Zowe API Mediation Layer

- ZAAS Client can now use HTTP so that the Application Transparent Transport Layer Security (AT-TLS) can be used for communication to ZAAS. [#813](#)
- Implemented the logout functionality in ZAAS Client. [#808](#)

- Added a more helpful and actionable description to message ZWEAM511E, which occurs when API ML does not trust the certificate provided by the service. #818

Zowe App Server

- The `install-app.sh` script used to install App Server plugins can now be used without Node.js. If Node.js is not detected when the script is executed, this behavior will be automated. You can also force this behavior with the environment variable `INSTALL_NO_NODE=1`, such as in the following example:
 - `INSTALL_NO_NODE=1 ./install-app.sh ~/zlux-editor` #137
- ZSS is now automatically registered to the API Mediation Layer when both are present, using a static registration file. #208
- Additional environment variables are now supported, which provides more options for TN3270 for the `instance.env` configuration file #1176 while also allowing TN3270 host to be specified during installation configuration. #1125. The following new environment variables are now supported in `instance.env` #108:
 - `ZOWE_ZLUX_TELNET_HOST = string`
 - `ZOWE_ZLUX_SSH_HOST = string`
 - `ZOWE_ZLUX_TN3270_ROW = number`
 - `ZOWE_ZLUX_TN3270_COL = number`
 - `ZOWE_ZLUX_TN3270_MOD = numbers 2-5 as well as "dynamic" or other variations of the word`
 - `ZOWE_ZLUX_TN3270_CODEPAGE = ccsid number or string as seen in the ui`
- The Agent API now provides limited information without the need for authentication. Non-admins are able to view a subset of the information available to admins, specifically regarding the functionality of Zowe. Examples of the information available to non-admins are: OS architecture and environment variables for Zowe configuration such as the components used and the ports they are accessible on. #211
 - `/server/agent/environment (limited info)`
 - `/server/agent/services`
- The ZSS `/unixfile` API has been updated to include an option to force file content to be sent or received as a specific encoding. If not specified, the pre-existing behavior of automatically

choosing encoding based on tagging and file extensions will be used. #160

- The app server can now read and use keys, certificates, and certificate authorities contained with PKCS12 files. This is in addition to existing support for PEM-encoded files as well as z/OS keyrings. #244

Zowe CLI

The following enhancements were added to the **core CLI**:

- Added a `--pattern` option to the `zowe files list all-members` command. The option lets you restrict returned member names to only names that match a given pattern. The argument syntax is the same as the "pattern" parameter of the ISPF LMMLIST service. #810
- Added new options `--lrecl` and `--recfm` to the `zos-files create` command. Use these options to specify a logical record length and record format for data sets that you create. #788

Zowe Explorer

- Added the Allocate Like feature. #904
- Added the ability to disable/enable profile validation. #922
- Added the ability to access other profiles during profile validation. #953
- Grouped Favorites by profile for Datasets, USS, and Jobs. #168
- Once entered, datasets and members are displayed in uppercase. #962
- Updated the environment check for Theia compatibility. #1009

Zowe JES/MVS/USS Explorers

The following enhancement was added to the **Explorer UI Server**:

- Explorer UI Server is now published as separate pax and decoupled out of explorers. #37

The following features and enhancements were added to the **JES Explorer**:

- Moved explorer-ui-server out of explorers into new `shared` folder under Zowe Runtime Directory. Changed JES lifecycle start script to use new shared location. #207
- Added context menu entry for download JCL used to submit a job. #335
- Updated webpack to latest version, added `.npmrc` to specify npm registry as config. #222

The following features and enhancements were added to the **MVS Explorer**:

- Extracted out explorer-ui-server, changed MVS lifecycle start script and packaging script accordingly. [#151](#)
- Added loading icon to editor menu bar when opening a dataset's contents. [#291](#)
- Added `.npmrc` to specify npm registry as config.

The following features and enhancements were added to the **USS Explorer**:

- Extracted out explorer-ui-server, changed USS lifecycle start script and packaging script accordingly. [#100](#)
- Added `.npmrc` to specify npm registry as config.

Bug fixes

The following bugs were fixed.

Zowe API Mediation Layer

- Changed the default expiration time value for JWT token to 8h for consistency with the z/OSMF default. [#615](#)
- Reduced excessive and unhelpful log messages. [#672](#)
- Added the Base Path field in the API Catalog if one is available, which can override the Swagger Base Path. This causes the proper Base Path to be displayed in the event that the api doc is not populated properly. [#810](#)
- Removed overwriting of the Swagger Base Path, which resulted in malformed API routes when the base URL is shared among multiple services. [#852](#)
- API ML was previously not reporting SSL certificate errors when servers were unable to communicate. Now, if a SSLEException occurs, SSL certificate errors are reported. [#698](#)
- Fixed language in log messages for consistency. [#830](#)

Zowe App Server

In previous versions, the environment `arch` and `os` fields were incorrect. This has been fixed, and the updated response from `/server/agent/environment` service is [#213](#):

Zowe CLI

The following bug was fixed in the **FTP plug-in for Zowe CLI**:

- Fixed an issue where the `view spool-file-by-id` command retrieved incorrect contents. [#61](#)

Zowe Explorer

- Fixed USS renaming issues. [#911](#)
- Fixed the deletion of datasets issue. [#963](#).
- Removed errors in Favorites items caused by profiles that are created by other extensions. [#968](#)

Version 1.15.0 LTS (September 2020)

Notable changes

Keyring support

Prior to v1.15, the Zowe z/OS components were only able to use a certificate held in a USS Java KeyStore. In v1.15, the Zowe z/OS components can now use a certificate that is held in a z/OS keyring as described in [Configuring Zowe certificates in a keyring](#).

For more information about Zowe certificates, certificate authorities, trust stores, and how they are used by Zowe, see [Configuring Zowe Certificates](#).

Auto-Save plug-in data

Plug-in developers can now make use of the new autosave feature, which can automatically save state data based on what the developer intends to retain, at regular time intervals. This is to protect against client crashes, and in the case of a crash, the apps are reopened upon desktop login and restored with the saved state. This new capability furthers the larger goal of high availability and fault tolerance for all Zowe components.

Support for starting Zowe API ML without z/OSMF on your system

By default, the API Gateway uses z/OSMF as an authentication provider. With the release of Zowe 1.15 it is now possible to switch to SAF as the authentication provider instead of z/OSMF. So, if you want to securely run the Zowe API ML but your system does not have z/OSMF, simply select SAF as your authentication provider. For more information on how to switch to SAF, see [API Gateway configuration parameters](#).

New features and enhancements

The following features and enhancements were added:

Zowe API Mediation Layer

- The API Path Pattern now supports `serviceId` as the first element. This improves the consistency of the URL when processing through the Gateway or outside of the Gateway. [#688](#)
- The SAF Provider can now be used as a possible authentication provider. This removes the API ML dependency on z/OSMF for authentication enabling SAF to obtain the JWT. [#472](#)
- The Swagger URL is now provided for z/OSMF. This URL provides full documentation containing the Try It Out functionality if the z/OSMF version supports the Swagger endpoint. Alternatively, the URL provides the info endpoint to directly enable access to Zowe endpoints. [#665](#)
- The default configuration of API ML now supports character encoding. [#777](#)

ZSS

A new endpoint has been added to the Agent API. This new endpoint will return a list of services to the user. [#209](#)

- Sample request: `GET /server/agent/services`
- Sample response:

Zowe App Server

- Added a feature that allows users to auto save plug-in data by subscribing to the event. By default, the feature will auto save every 5 minutes, but this interval can be customized. [#250](#)
 - This feature is enabled via the Plugin Definition. `"autosave": true`
- You are now able to select multiple jobs in the job tree, which allows for functions such as purging multiple jobs at once. [#274](#), [#204](#)

Zowe CLI

The following features and enhancements were added to the **core CLI**:

- Added a `--responseTimeout` option to the z/OS Files APIs, CLI commands, and z/OSMF profiles. Specify `--responseTimeout <###>` to set the number of seconds that the TSO servlet request runs before a timeout occurs. The default is 30 seconds. You can set the option to 5 - 600 seconds (inclusive). [#760](#)
- Added the `--encoding` option for the `zowe zos-files upload dir-to-pds` command. This option lets you upload multiple members with a single command. [#764](#)

The following features and enhancements were added to the **Imperative CLI Framework**:

- Added support for dynamically generated cookie names. Updated `AbstractSession.storeCookie()` to process cookie names that are not fully known at build-

time. [#431](#)

- Added the SSO Callback function, which allows applications to call their own functions while validating session properties (that is, host, port, user, password, token, and so on). The callback option is named `getValuesBack`. [#422](#)

The following features and enhancements were added to the **Secure Credential Store Plug-in**:

- Added the `scs revert` command. Use the command to revert securely stored credentials in your user profiles to be stored in plain text. [#22](#)
- Changed the `scs update` and `scs revert` commands so that they fail if Secure Credential Manager is not enabled. [#23](#)

Zowe JES/MVS/USS Explorers

The following features and enhancements were added to the **JES Explorer**:

- Changed the packaging and lifecycle `start.sh` script to add explorer-ui-server keyring support. [#1177](#)
- Added app bar, along with settings, and local storage to store user preferences and remember the last search filter. [#487](#)
- Notifications preference can set duration for snack bar notification. [#273](#)

The following features and enhancements were added to the **MVS Explorer** and **USS Explorer**:

- Changed the packaging and lifecycle `start.sh` script to add explorer-ui-server keyring support. [#1177](#)
- Added ability to collapse and resize jobs tree. [#259](#)

Bug fixes

The following bugs were fixed.

Zowe API Mediation Layer

- Fixed SSL validation when Eureka is running in HTTP mode. When the scheme is HTTP, SSL configuration is not verified since it is not used. [#792](#)
- Fixed a problem in error handling when no api-doc is available. Now a specific return code and message is generated when a problem occurs when obtaining or transforming the api-doc. [#571](#)

- When RBAC is disabled, only the following services will be available. #210
 - `/server/agent/environment` (with limited information)
 - `/server/agent/services`

Zowe App Server

- External CA certificates to the Zowe `ZWED_node_https_certificateAuthorities` array only after checking to see if the certificates exist, which prevents it from pointing to nothing, resulting in it breaking. #136
- In previous versions, the `component.json` file was only being created when users upgraded their Zowe system to a more recent version. Performing an initial installation would not result in the `component.json` file being created. In this version, this bug has been resolved, and the `component.json` file is created both when upgrading and performing an initial installation. #135

Zowe CLI

The following bugs were fixed in the **core CLI**:

- Renamed the z/OS Files API option from `storeclass` to `storclass`. This fixed an issue where the CLI could define the wrong storage class on `create dataset` commands. #503
- Fixed an issue where the output of the `zowe zos-uss issue ssh` command would sometimes omit the last line. #795

The following bug was fixed in the **Imperative CLI Framework**:

- Fixed an issue with `ConnectionPropsForSessCfg` where the user would be prompted for user/password even if a token was present. #436

Zowe JES/MVS/USS Explorers

The following bugs were fixed in the **JES Explorer**:

- Fixed a bug where no jobs would show after auth token expired and user logs back in. #408
- Added default value for `ZOWE_EXPLORER_FRAME_ANCESTORS` at lifecycle start script. It resolves #44.
- Fixed an issue where job tree height is greater than app container which makes the page scrollable. #484

The following bugs were fixed in the **MVS Explorer**:

- Fixed an issue where the dataset tree and the content viewer were not aligned. #484

- Added default value for `ZOWE_EXPLORER_FRAME_ANCESTORS` at lifecycle start script. It resolves #44.

The following bugs were fixed in the **USS Explorer**:

- Added default value for `ZOWE_EXPLORER_FRAME_ANCESTORS` at lifecycle start script. It resolves #44.

Version 1.14.0 LTS (August 2020)

Notable changes

Zowe Node APIs

Did you know that you can leverage the Zowe Node APIs directly? The Zowe Node APIs are the programmatic APIs that enable Zowe CLI to interface with the mainframe. You can use the APIs to build your own applications or automation scripts, independent of Zowe CLI. For more information and usage examples, see the [Zowe CLI readme file](#).

Support for verifying Zowe release integrity

Zowe now provides a new tool to verify that the code in the Zowe runtime directory installed on your z/OS® system is identical to the released code. The tool comprises a script file `zowe-verify-authenticity.sh`, plus the files it needs to check the release contents.

If the contents of the Zowe runtime directory have been modified, then it may result in unpredictable behavior. For more information about the tool, see [Verify Zowe runtime directory](#).

New features and enhancements

The following features and enhancements were added.

Zowe installation

- If you are upgrading to Zowe v1.14 from a previous release, and the value of `ZOWE_EXPLORER_HOST` does not match the host and domain that you put into your browser to access Zowe, you must update your configuration due to updated referrer-based security. See [System Requirements](#) for information on updating your configuration.
- Allow the user to verify the authenticity of a Zowe driver. The script `zowe-verify-authenticity.sh` will check that a Zowe `ROOT_DIR` for an installed release matches the contents for when that release was created, which assists with support and troubleshooting. To verify pre-

1.14 releases, the script and its associated code are available [separately](#) (see #1552). For more information, see the new topic [Verify Zowe Runtime Directory](#) that describes the operation of the script.

- Allow multiple domains (names/IP Addresses) when generating certificates. This also includes SMP/E `HOLDDATA` for the affected function [Zowe Configuration](#). #1511
- Included z/OSMF workflows for Zowe z/OS configuration. #1527
- Added warning if `ZWESVSTC` runs under user ID `IZUSVR`. #1534
- [Docs] Changed the documentation so that `SZWEAUTH` PDSE load library members should not be copied elsewhere, but instead the original installation target `SZWEAUTH` PDSE should be APF-authorized and used as the runtime load library. This also includes SMP/E `HOLDDATA` for the affected function `STC JCL` as well as changes to topics [Installing and configuring the Zowe cross memory server \(ZWESISTC\)](#) and [Installing and starting the Zowe started task \(ZWESVSTC\)](#).
- [Docs] Added a new topic [Installing and configuring Zowe z/OS components using scripts](#).

API Mediation Layer

- Prevented crashing of API ML when null routes are set. #767
- Added support to the X-Forwarded-* Headers. #769
- Improved the configuration validator for the enablers to improve message specificity when one or more parameters required for setup are missing. #760

Zowe App Server

- Using a cross-memory server without `REUSASID=YES` may result in an ASID shortage. This pull-request adds a check that will print a warning if `REUSASID=YES` is not detected. #145
- In previous versions, the server used the property `InstanceID` instead of `ZOWE_INSTANCE`. In order to maintain backwards compatibility, these properties are now unified when the value of `ZOWE_INSTANCE` is non-default. Additionally, the server uses these values whenever an instance number is needed, such as in the case of determining profile names for RBAC use #130
- The packaged size of the Editor has been significantly reduced by removing uncompressed versions of files that have compressed variants and `.map` files which were used for development debugging. #160
- The ZSS /unixfile REST API now supports the changing of permissions on a file or folder, similar to `chmod`, by calling `/unixfile/chmod`. The behavior is documented [in swagger](#). #195 #132
- A notification will be displayed when users attempt to upload a wallpaper image that is too large. #254

- The desktop personalization panel's color selection UI now has an extra highlight around the selected color to make the selection more apparent. [#236](#)
- Users can now recall migrated datasets in the Editor (via the File Tree) by clicking on them. [#78](#)

Zowe CLI

The following features and enhancements were added to the **core CLI**:

- Added the command `zowe zos-files delete migrated-data-sets` to delete migrated data sets. [#716](#)
- Added a new `--fail-fast` option to the `zowe zos-files download all-members` command. The option defaults to `true`, which preserves existing behavior. Set the option to `false` to continue downloading members if one or more of the downloads fails. [#759](#)
- Updated the Imperative CLI Framework version. [#744](#)

z/OS FTP Plug-in for Zowe CLI:

The following enhancement was added to the z/OS FTP Plug-in:

- The following flags were added to the `zowe zos-ftp submit data-set` command: [#55](#)
 - `--wait` - Specify a query interval and max times to query as comma-separated, numeric values. For example, specify `5,12` to query the job status every 5 seconds up to 12 times.
 - `--wait-for-output` - Wait for the job to enter OUTPUT status.
 - `--wait-for-active` - Wait for the job to enter ACTIVE status.

Zowe Explorer

The following features and enhancements were added to the **Zowe Explorer**:

- Added a webpack that works with localization and logging.
- Allowed extenders to load the saved profile sessions upon activation.
- Added an automatic re-validation for invalid profiles.

Also, check out [the Zowe Explorer FAQ](#) to learn more about the purpose and function of the VS Code extension.

Bug fixes

The following bugs were fixed.

Zowe App Server

- Bugfix: ZSS will now maintain the connection if users respond to the 404 message with the request `Connection: Keep-Alive` [#147](#)
 - **NOTE:** The code only recognizes `Connection: Keep-Alive`. Other "Keep-Alive" properties will be ignored.
- Bugfix: If a load module is incorrectly copied to STEPLIB, the z/OS loader will fail to load it. In these cases, an available copy in LPA will be used instead, if one is available. The problem with LPA is that any IDENTIFY calls to a module with an incorrect version number may cause serious issues. This pull-request ensures that ZWESIS01 comes from private storage. [#146](#)
- Bugfix: Fixes various issues that would occur when the number in the `Content-length` response header was different from the actual content length. [#150](#)
- Bugfixes for default plugin config and terminal handler location. This change was made in order to include the `_internal` folder. `storageDefaults` other than `_internal` are already supported. For more information, see the [wiki](#). [#229](#)
 - This fix allows the server-side plugin config to exist within its own folder, rather than in the instance directory. As a result, plugins no longer have to perform a copy operation during installation.
 - You can now specify terminal proxy handler overrides within `$INSTANCE_DIR`, which was previously only possible within `$ROOT_DIR`. `$ROOT_DIR` modification is not recommended and not conformant for Zowe plugins.
- Bugfix: The process of auto-converting untagged USS ebcDIC files when using the ZSS /unixfile REST API has been improved by determining if the files are text or binary based on a list of file extensions. The API behavior towards unknown extensions has been changed from assuming text to now assuming binary. This fixed some cases where text files were not readable through the REST API. [#148](#) [#152](#)
- Bugfix: When using ZSS's /unixfile/contents REST API, large files would occasionally cause an incorrect HTTP message to be sent because the content-length header did not match the actual content length. This could result when there is a conversion error. This issue has been solved by updating the API, allowing it to use the transfer encoding type "chunked" instead, which allows these previously broken files to be sent successfully. [#150](#)
- Bugfix: Some file actions in the Editor would generate URLs that included multiple slashes in a row, which may cause errors on servers that receive such requests. In this update, the URI Broker now removes multiple slashes when they are encountered, which may additionally improve behavior in other apps that use the URI Broker for the ZSS REST API /unixfile. [#251](#)
- Bugfix: During ZSS initialization, certain warning log messages were not displayed, such as the warning about lack of permission to use ICSF to generate a random number. This issue has been

resolved by initializing the logger responsible for issuing the messages. [#143](#)

- Bugfix: In order to conserve log space, ZSS no longer prints debug information regarding HTTP dispatch. [#156](#)
- Bugfix: In previous versions, the app framework build process referenced webpack incorrectly, leading to an unnecessary build-time error if webpack was not installed globally. This issue has been resolved. [#248](#)
- Bugfix: In previous versions, developing with the app framework would show linting warnings in VSCode. This issue has been resolved by updating tsconfig.json [#240](#)
- Bugfix: Some app server configuration values could not be specified via environment variables due to the limited characters allowed in variables. A new syntax has been made to allow these edge-case configuration values to be specified, and this new syntax is seen here: [#230](#)
 - Overall behavior is described [in the wiki](#).

Zowe CLI

The following bug was fixed in Imperative CLI Framework:

- Fix update profile API storing secure fields incorrectly when called without CLI args.
- Fixed a compilation error when building the CLI from source. [#770](#)

Zowe Explorer

- Fixed the bug related to saving USS files.
- Fixed the bug related to the deletion of datasets.

Version 1.13.0 LTS (July 2020)

Notable changes

Zowe CLI added the ability to access mainframe services through API Mediation Layer using single-sign on (SSO) and multi-factor authentication (MFA). Use Zowe CLI to log in to API Mediation Layer and receive a token that is used for secure authentication to one or more services. For more information, see [Integrating CLI with API Mediation Layer](#).

The CLI also supports a type of profile named "base profile" that lets you store configuration information for multiple services. For more information, see [Using Profiles](#).

New features and enhancements

The following features and enhancements were added.

Zowe installation

- Updated `zowe-configure-instance` upgrade to update ROOT_DIR. This allows you to move the Zowe runtime to a different place when you install a new version of Zowe. [#1414](#)
- Updated the port validation logic to reduce false negatives. [#1399](#)
- Updated the Zowe installation and configuration to tolerate ZERT Network Analyzer better. [#1124](#)

API Mediation Layer

- Added Cross-origin resource sharing (CORS) Headers Support.
- Introduced an option to set connection timeout for a service.
- Provided SAF Keyrings support for a ZAAS Client.
- Introduced Spring Boot enabler configuration validation.

Zowe App Server

- The app server is now able to use more than one certificate authority (CA). This allows the server to validate other server's authenticity by recognizing the CA that another server may have used [#128](#)
- The `dispatcher.invokeAction` method now returns promise, which provides the ability to wait until `dispatcher.invokeAction` finishes and handles errors [#59](#)
- The ngx-color picker has been replaced by a custom hue selection bar, lightness swatches bar, and color palette, allowing for a more customizable personalization experience [#235](#)
- In this version, cross-launch via URL has been implemented, allowing for integration between the Application Framework and applications. This feature enables users to bookmark a set of app2app communication actions (in the form of a URL) that will be executed when opening the webpage [#234](#)
- Bookmarking features have been added to the TN3270 emulator [#30](#)
 - Users can now save connection preferences on a per-user level. Clicking the floppy disk icon saves user settings to that user's scope.
 - Codepages have been reorganized so that the numbers are shown first, making it easier for users to navigate to their favorites
 - The buttons found in this feature have been realigned
- Several features have been added to the Zowe Editor [#153](#)
 - Globally increased the shortest duration of snackbar notifications from 2 seconds to 3 seconds

- Added a "Close All" button in the menu (hot key is Alt + W + Shift)
- A snack bar notification will be displayed when users attempt to open a file that they do not have permission to open
- Added an "Undo" option to the Close All feature to reopen tabs & files
- Login activity and session activity is now synchronized across multiple desktop tabs [#242](#)
 - When a user logs out of a desktop tab, all other active tabs will also log out
 - When a user performs an action on a desktop tab, the other tabs register this activity, which stops them from timing out

Zowe CLI

The following features and enhancements were added to the **core Zowe CLI**:

- Added the ability to log into and out of API ML using a token. [#718](#)
- Added the `--base-profile` option to all commands that use profiles to let them make use of base profiles that contain shared values. [#718](#)
- CLI commands now prompt for any of the following option values if the option is missing: host, port, user, and password. [#718](#)
- Added character encoding/code page support for download and upload data set operations in the API library and the CLI. [#632](#)
- Added the `--encoding` option to the `zosmf` profile type. [#632](#)
- Introduced an API to delete migrated data sets. [#715](#).

The following features and enhancements were added to the **Imperative CLI Framework**:

- Added the `ConnectionPropsForSessCfg.addPropsOrPrompt` function to store credentials, such as a token, in a session configuration object. [#718](#)
 - CLI plug-ins must implement this function to create sessions in order to consume automatic token-handling and prompt for mission options features.
 - Connection information is obtained from the command line in the following order: Environment variables, service profiles, base profiles, or a default option value.
 - If connection information is not supplied to any core CLI command, the user is prompted for:
 - host
 - port
 - user

- password

The prompt times out after 30 seconds so that automated scripts will not fail.

- Added base profiles, a type of profile that can store values and provide them to other profile types, such as zosmf profiles. [#402](#)

The following properties can be stored in a base profile:

- host
 - port
 - user
 - password
 - rejectUnauthorized
 - tokenType
 - tokenValue
- Added `login` and `logout` commands to retrieve and delete tokens. [#405](#)
 - Added a `showToken` flag to display the token and not save it to the user profile.
 - Added the ability to create a user profile upon login, if no profile of that type existed previously.
 - Added the `--dd` flag, which lets users create a profile without using the default values specified for that profile. [#718](#)
 - If a token is present in the underlying REST session object, Imperative uses the token for authentication.
 - CLI help text includes new options such as `tokenValue`. Plug-in developers might need to update mismatched snapshots in automated tests.
 - Updated the version of TypeScript from v3.7.4 to v3.8.0.
 - Updated the version of TSLint from v5.x to v6.1.2.
 - Update log4js to improve Webpack compatibility for extenders.

Zowe Explorer

The following features and enhancements were added to **Zowe Explorer**:

- Added a credentials check feature that allows users to update their credentials if they receive an authorization error.
- Added a star icon that clearly denotes data sets, USS files, and jobs as favorites.
- Added a profile validation feature that checks whether a profile is valid. The feature is triggered when any action is performed with the profile. Validated profiles are indicated by a green mark.
- Disallowed case sensitivity for profiles with same names.
- Enabled editing of search filters.
- Enabled editing of ASCII files in USS.
- Improved text in confirmation dialogs.
- Reorganized the Data Sets context menu to match the order of commands recommended by VSCode.

Bug fixes

The following bugs were fixed.

ZSS

- Bugfix: ICFS error message is not printed. In this version, the issue has been resolved [#143](#)

Zowe App Server

- Bugfix: Changing editor syntax in the MVS explorer caused a callstack limit exception. This was due to a trap focus conflict between the Orion editor and the modal part within the ui Select component on syntax change. In this version, the issue has been resolved by disabling `disableEnforceFocus` for the syntax selector [#129](#)
- Bugfix: An Infinite Auth loop would occur on explorer apps due to APIML and z/OSMF auth timeouts mismatch. In this version, the issue has been resolved by adding a force login flag if a datasets request comes back as 401 [#124](#)
- Bugfix: When using the JES Explorer to view Spool files of a job, users cannot open a spool file that has the same name as one already open. This issue has been resolved by adding a unique id to content tabs to allow opening of overlapping names [#188](#)
- Bugfix: The `Env var` for `TERM` gets set to "linux", which is not recognized by USS. This issue has been resolved through the removal of rxjs-compat [#29](#)
- Bugfix: NGX-monaco-editor library has been removed in order to fix a bug. This now allows the Editor to open and view files after the second instance of opening them [#155](#)
 - Removed use of node-sass, so that native compilation is not required
 - Updated to typescript 3.7 from version 2.7.2

- Updated to monaco 0.20 from version 0.13. The monaco changelog can be found [here](#)

Zowe CLI

- Fixed an issue where CLI web help failed to load in Internet Explorer 11. [#393](#).
- Fixed an issue where the `--help-web` option did not function on macOS when the `DISPLAY` environment variable was undefined. [#322](#).
- Updated Imperative version to include security fixes.
- Updated Imperative version to fix a problem where users could not use a service profile after storing a token in a base profile.
- Fixed an issue where optional secure fields were not deleted when overwriting a profile.

Version 1.12.0 LTS (June 2020)

New features and enhancements

The following features and enhancements were added.

Zowe installation

- Keystore directory generation updated to add new parameters. If you wish to enable SSO for the desktop you need to rerun the `zowe-setup-certificates.sh` script during the upgrade process, with new values in the `zowe-setup-certificates.env` file. [#1347](#) / Doc: [#1162](#)
- Added a `-l` optional parameter to the `zowe-support.sh` script. This parameter allows you to specify the custom log directory used in installation and configuration when collecting support data. [#1322](#) / Doc: [#1165](#)
- Added the validate only mode of Zowe. This allows you to check whether all the component validation checks of the Zowe installation pass without starting any of the components. [#1335](#) / Doc: [#1181](#)
- Separated ZSS component from the Zowe App Server component. [#1320](#)
- Introduced z/OSMF Workflows that accomplish the following Zowe installation and configuration tasks:
 - Install Zowe runtime using z/OSMF Workflows.
 - Configure z/OS Security Manager.
 - Configure Zowe certificates.
 - Create and configure the Zowe instance directory, and run the Zowe started task.

API Mediation Layer

- Provided Zowe Authentication and Authorization Service (ZAAS) client.
- Refreshed the static client definitions from the API Catalog UI.
- Switched to `sso-auth` instead of `apiml-auth`.
- Added logout endpoint API documentation.
- Made `jjwt` only a test dependency.
- Fixed the order of fetching the JWT from a request.
- Implemented request retrying for service instances.

ZSS

- ZSS now follows the Zowe Component scheme, as part of the DESKTOP component group [#177](#)
- Read JWT token information from environment variables, if they exist, to further support SSO during a standard installation. [#178](#)
- In previous versions, ZIS did not use the version information provided in `zss/version.txt`. In this version, the ZIS build uses `version.txt` the same way that ZSS uses it. [#184](#)

Zowe App Server

- Added SSO token name and label to `convert-env.sh` for use with ZSS. [#118](#)
- Script has been updated to allow ZSS to be a separate component. [#117](#)
- The app-server will favor and use a SAF keyring if defined for use in Zowe, rather than a unix file for keys, certificates, and certificate authorities. [#116](#)
- The process for making bundled plugins using `ROOT_DIR` has been upgraded [#123](#)
- Updates have been implemented for modal keyboard accessibility. [#148](#):
 - Editor now has keyboard navigation in the browsing tree and pop-up modals.
 - Pop-ups can be traversed with Tab/Tab + Shift.
- Desktop redesign suite and personalization settings have been implemented. [#221](#)
- Right-click context menus have been implemented for the new desktop style. [#216](#)
- A new attribute has been implemented to load plugins from different relative paths. [#212](#)

Zowe CLI

The following features and enhancements were added to the **core Zowe CLI**:

- Added the `zowe files hrec ds` command to recall data sets. [#556](#)
- Made the `account` option optional in TSO profiles. [#709](#)

- Made `user` and `host` options optional in SSH profiles. [#709](#)

The following features and enhancements were added to the **z/OS FTP Plug-in for Zowe CLI**:

- Added the `zowe zos-ftp list data-set-members` command to find members in a PDS. [#45](#)
- Added the `zowe zos-ftp make uss-directory` command. [#47](#)

Zowe Explorer

Review the [Zowe Explorer Change Log](#) to learn about the latest features, enhancements, and fixes.

You can install the latest version of the extension from the [Visual Studio Code Marketplace](#).

Bug fixes

The following bugs were fixed.

Zowe installation

- Minor enhancements to add log directory validation and remove unnecessary log file splitting. [#1334](#), [#1300](#)
- When the automatically detected hostname that Zowe is installed on cannot be resolved, use the IP address instead. This covers the scenario when the USS `hostname` command returned a system name that wasn't externally addressable. [#1279](#)
- Fixed an issue that could cause an upgraded version of Zowe to try and use an old version of plugins, by switching the desktop to use a relative reference to find plugins. [#1326](#)

ZSS

- Bugfix: Fixed a segfault when no config file is provided by moving all the zowelog invocations to a location where the logging environment is ready. Additionally, cleanup logic has been introduced to ensure that we free the STC base resources before leaving main. [#187](#)
- Bugfix: In previous versions, if a warning message is produced by the compiler, the build process is considered successful. This is often dangerous as warnings can indicate passing the wrong type or redefinition of a `#define`, which should be considered bugs. The following changes have been implemented to make the build process more strict [#188](#):
 - Make sure there are no warning messages in the current build.
 - Update deps to remove the `httpfileservice.c` warning message, and pick up a minor type fix.
 - Ensure side-deck `file/SYSDEFSD DD` by adding the `dll` option to the linker.

- Adjust the compiler env variable that controls the severity.
- Ensure no ZSS binary is created if RC != 0.

Zowe App Server

- Bugfix: Logout of sso-auth was not working because it was expecting apiml parameters that should have been there but were controlled by the env var APIML_ENABLE_SSO. In this version, the issue has been resolved. [#126](#)
- Bugfix: In this release, many bugs picked up by the Sonar scan for core Zowe repositories have been resolved [#214](#)
- Bugfix: Plugin api would not respond if a plugin could not load due to a dependency not being met. That plugin would not be placed in the array that checks when the processing has finished, so a response would never be generated. [#208](#)
- Bugfix: Fixed a logout cookie bug and sso-auth behavior bug in order to fully support SSO. Additionally, `tokenInjector` was removed as it is no longer required with the introduction of SSO. [#209](#)
- Bugfix: Fixed lease information for API ML [#218](#)
- Bugfix: In previous versions, the user was never shown the logout screen when the plugin would detect zss, but not apiml. In this version, this issue has been resolved. [#221](#)
- Bugfix: Fixed issue where localhost & 127.0.0.1 were always used even when not true. Additionally, each worker in the cluster attempted registration even though, from an outside perspective, it is 1 server. In this version, the server uses a real hostname and tries to find the ip that best matches what apiml would be able to use [#203](#)

Zowe CLI

Updated Yargs in Zowe Imperative CLI Framework to fix vulnerabilities.

Version 1.11.0 LTS (May 2020)

New features and enhancements

The following features and enhancements were added:

API Mediation Layer

The following new feature was added to the Zowe API Mediation Layer in this version:

- The 'Try it out' functionality has been added to test for public and private endpoints. [#258](#)

ZSS

- A new query parameter (?addQualifiers) which can be appended to /datasetMetadata/ allows for searching that more closely represents the search behavior of 3.4 [#108](#)
- Added support for changing log levels via REST API [#173](#)

Zowe App Server

- Updated the JES Explorer, MVS Explorer, and USS Explorer apps to support single sign-on from the Zowe API Mediation Layer. [#344](#) [#345](#) [#346](#)
- Modals in the Editor now have an "X" icon to close the modal. [#130](#)
- An event emitter for session changes, login, logout, and sessionExpire for Angular, React, and iFrame applications has been added [#210](#)
- Session events have been added to mvdhosting [#53](#)
- Updates made to generate_zlux_certificates.sh because apiml_cm.sh has been moved into the zowe-install-packaging repo [#110](#)
- Zowe Web browser plugin, which can be used to view webpages that are not Zowe apps, has been added. [#194](#)
- Translations have been added for labels and buttons for password reset forms [#215](#), [#218](#)
- Browser-based apiml token, auth simplification [#196](#):
 - i. API mediation layer token is now held in the browser upon login via the Desktop. This also allows for the Desktop to do single-sign-on login with the token if it is already present in the browser.
 - ii. Auth plugins no longer need to be specified explicitly within the server configuration file, the capability remains for backwards compatibility. The server will now auto-detect the auth plugins that are available
 - iii. Auth plugins can now be of more than one type, to satisfy environments that have plugins that need access to APIs of similar but different types
- New shortcuts have been added to navigate the start menu with a keyboard [#213](#)
- Sessions are now maintained based on most recent activity across tabs [#219](#)
- Support for password changing, including expired password changing, has been implemented [#193](#)

Zowe APIs

Zowe Jobs APIs

- Version 2 APIs now support single sign-on from the Zowe API Mediation Layer [#21](#)
- Updated embedded Spring Boot version [#89](#)

Zowe Data Set and Unix Files APIs

- Version 2 APIs now support single sign-on from the Zowe API Mediation Layer [#18](#)
- Updated embedded Spring Boot version [#151](#)
- Added incomplete connect timeout parameter to prevent Slowloris DOS attacks [#158](#)

Zowe CLI

Reference the appropriate version in each of the following changelogs to learn about CLI features, enhancements, and fixes:

Core CLI Changelogs:

- [Zowe CLI - v6.11.0](#)
- [Imperative CLI Framework - v4.6.0](#)
- [Secure Credential Store Plug-in - v4.0.4](#)

CLI Plug-in Changelogs:

- [IBM CICS Plug-in - v4.0.2](#)
- [IBM DB2 Plug-in - v4.0.6](#)
- [IBM FTP Plug-in - v1.0.2](#)
- [IBM IMS Plug-in - v2.0.1](#)
- [IBM MQ Plug-in - v2.0.1](#)

Zowe Explorer

Review the [Zowe Explorer Change Log](#) to learn about the latest features, enhancements, and fixes.

You can install the latest version of the extension from the [Visual Studio Code Marketplace](#).

Zowe installer

- Added a `-l` parameter to the `zowe-install.sh`, `zowe-setup-certificates.sh`, `zowe-install-xmem.sh`, and `zowe-install-proc.sh` scripts. This parameter allows you to specify where the setup scripts write trace logs.
- Improved port validation to assist determining whether Zowe's ports are available.

Zowe troubleshooting

- Improved the troubleshooting script `zowe-support.sh` to assist with offline problem determination. See [Capturing diagnostics to assist problem determination](#).

Zowe documentation

- Added a topic [Zowe runtime lifecycle](#) that describes the use of the `EXTENDER_COMPONENTS` value in the `instance.env` file. See [Creating and configuring the Zowe instance directory](#).
- Improved the [Zowe architecture](#) information to include a more current architecture topology diagram and more details on the individual Zowe services, where they log their data, and how to perform high-level problem determination.
- Added new problem determination scenarios and resolution. See [Troubleshooting Zowe Application Framework](#)
- Added information on how to determine which release of Zowe is installed. See [Zowe releases](#).
- Added a [Zowe resources](#) topic, which provides a list of resources that supplement the documentation on this site.

Bug fixes

The following bugs were fixed:

ZSS

- Bugfix: Fixed a below-the-line leak in the QSAM code [#138](#)

Zowe App Server

- Bugfix: Material dialogs no longer overlap over the login screen [#145](#)
- Bugfix: Re-login to same desktop session would duplicate items in the launch menu. In this version, the session is cleared on logout, fixing the duplication issue [#208](#)
- Bugfix: Bugfix for websockets to prevent server throwing exception on malformed message [#189](#)
- Bugfix: Fixed app server configuration bug where min worker count was ignored when max worker count was not defined [#187](#)
- Bugfix: Added missing pluginID argument for setStorageAll method. [#191](#)
- Bugfix: app-server agent information was not available to plugins if it was specified via command line arguments [#111](#)

Version 1.10.0 LTS (April 2020)

New features and enhancements

The following features and enhancements were added:

API Mediation Layer

The following new feature was added to the Zowe API Mediation Layer in this version:

- Zowe API ML can now use z/OSMF to provide JSON Web Tokens (JWT). [#433](#)

ZSS

- Fast EBCDIC to UTF8 character translation is now supported by using the TROO instruction with a "EBCDIC 1047 to ISO/IEC 8859-1" translation table. [#127](#)
- Performance improvements in character conversion, JSON and collections code. [#162](#)
- The code now prints fewer warnings when AT-TLS is not set up. [#130](#)
- ZSS logs belonging in the ZSS repo have been refactored so that they now use the Zowe logger and message IDs. [#163](#)
- Config variable names have been updated to stay consistent with IBM terminology. [#165](#)

Zowe App Server

- The sample-react-app README has been updated to state prerequisites. [#20](#)
- An example of how to use the Zowe Desktop's built-in context menu has been added. [#31](#)
- Sample angular app has been updated for angular 6 best practices use of HttpClient, RxJS [#33](#)
- Simple conda build scripts have been added. [#46](#)
- App server logs now have IDs prefixed, for easy lookup in future documentation. [#49](#)
- Enhancements for plugin adding. [#51](#)
- App server logs now have IDs prefixed, for easy lookup in future documentation [#102](#).
- App server now defaults to prevent apps from being embedded in an iframe that does not come from the same origin. [#104](#)
- The jes-explorer has been updated to support Single Sign On functionality offered by the api-layer. [#160](#)
- Desktop now has key bindings to minimize (ctrl-alt-down), maximize windows (ctrl-alt-up), and show launchbar menu (ctrl-alt-m). [#176](#)

- App server “router”-type dataservices now have a new Storage API within their context object, for standardized in-server state persistence. [#178](#)
- App server can now add plugins on-demand without a restart, by re-scanning plugins directory via REST API /plugins. [#179](#)
- App server can now be configured to set HTTP headers that will default and possibly override those of the plugins. [#180](#)
- App server /auth API now returns which handler is the default. [#183](#)
- Events and actions for viewports and windows are now accessible to iframe via the standardized window.ZoweZLUXiframe object. [#184](#)
- Focus on app2app, as well as some package updates. [#188](#)
- 3 features:
 - i. Desktop can now filter the list of apps by search query.
 - ii. Desktop cleanup has reduced the bootstrapping server requests by half.
 - iii. Desktop now can load new apps added to the server without a page reload. [#189](#)
- Desktop’s DOM now has lang attribute as soon as the language preference is known. [#190](#)
- Desktop login screen updated with new Zowe logo. [#204](#)
- JES, MVS Explorers now have support for APIML’s Single Sign On feature [#344](#)

Zowe CLI

The Secure Credential Store plug-in is now packaged with tools that build dependencies locally. This fixes an issue where the installation could fail at sites with firewall restrictions. [#9](#)

Tip: Zowe CLI release notes are now aggregated in changelogs. Reference the appropriate version in each changelog to learn about features, enhancements, and fixes.

Core CLI Changelogs:

- [Zowe CLI - v6.10.1](#)
- [Secure Credential Store Plug-in - v4.0.3](#)

CLI Plug-in Changelogs:

- [IBM CICS Plug-in - v4.0.2](#)
- [IBM DB2 Plug-in - v4.0.5](#)
- [IBM FTP Plug-in: - v1.0.1](#)
- [IBM IMS Plug-in: - v2.0.1](#)
- [IBM MQ Plug-in: - v2.0.1](#)

Zowe Explorer

Review the [Zowe Explorer Change Log](#) to learn about the latest features, enhancements, and fixes.

You can install the latest version of the extension from the [Visual Studio Code Marketplace](#).

Bug fixes

The following bugs were fixed:

Zowe z/OS Installation

Bugfix: `zowe-configure-instance.sh` does not allow the `-c` instance directory location to be an existing Zowe runtime. This caused a deadlock and running out of `BPXAS` instances. See [Unable to create BPXAS instances. #1123](#)

Zowe App Server

- Bugfix: subloggers would not inherit message translation maps from parent loggers. [#24](#)
- Bugfix: sample-angular-app could not be run from a folder outside of `$ROOT_DIR`. [#34](#)
- Bugfix: Menu locations were wrong when multiple apps opened because the numbers used partially came from the previous instance. [#36](#)
- Bugfix: Apps that were the target of app2app communication were not put into focus. [#50](#)
- Bugfix: Developers could not run app-server without a certificate authority. [#98](#)
- Bugfix: App server could not work with self-signed/invalid TLS certificates sometimes used in test/development, because the configuration option broke. The option has been restored. [#103](#)
- Bugfix: App server instance settings initialization had inconsistent write permissions. [#105](#)
- Bugfix: App server no longer issues warning about failure to load undefined log file. [#182](#)
- Bugfix: Fixes unformatted messages when a language is not specified. [#186](#)
- Bugfix: Editor would not work for unix files when used through api mediation layer due to encoded slash. [#187](#)
- Bugfix: App framework's right click menu could go off screen vertically at the bottom. [#200](#)
- Bugfix: zosmf-auth no longer issues configuration warning during startup. [#398](#)
- Doc Bugfix: Sample react app did not state its dependence on the sample angular app. [#405](#)
- Bugfix: Substitute zosmf-auth for apiml-auth to remove warning. [#1232](#)

Version 1.9.0 LTS (February 2020)

Zowe v1.9.x is designated as the current Zowe Long-term Support (LTS) version.

New features and enhancements

The following features and enhancements were added:

API Mediation Layer

The following new features and enhancements have been made to the Zowe API Mediation Layer in this version:

- Support of special characters has been added to API Mediation Layer core services. In addition, all onboarding enablers now support special characters as well.
- Custom metadata support has been added to the onboarding enablers. Additional parameters can now be easily added to an expandable parameter array. This feature may be used for security configuration in the future.
- Passticket support has been added to API ML Core Services and onboarding enablers. This makes it easier to authenticate existing mainframe applications with the API Mediation Layer.
- New versions of Spring Boot based onboarding enablers (V1 and V2) have been released. These enablers support the new version of the metadata required by the Discovery Service. The new versions of the enablers consume significantly less disk space.

The following bug fixes have been introduced:

- A fix of a critical authentication issue with some versions of z/OSMF has been applied.
- A fix has been applied to support multipart requests.
- A fix has been applied to the z/OSMF authorization header.

Zowe App Server

- Added support for Node.js - z/OS V12. See [Installing Node.js on z/OS](#) for details.
- A new endpoint for removing dataservices has been added [#62](#)
- Functionality for removing data sets has been added [#65](#)
- Deletion of data sets and their members is now supported [#88](#)
- Deletion of data sets and their members is now supported [#85](#)
- The following helper functions have been added to test caller's environment [#115](#):
 - A function to test whether the caller is running in SRB
 - A function to test whether the caller is in cross-memory mode
 - A function to test whether the caller is holding a CPU, CMS, CML or local lock

- The logout endpoint has been re-added for zss [#100](#)
- Added support of SRB and locked callers to the Cross-Memory server's PC space switch routine [#153](#)
- This pull request add the following features [#120](#): Ability to use the lock-free queue intrusively which allows a more flexible storage management on the user's side Functions to copy to/from foreign address space using destination/source keys and ALETs
- Reformatted the save as modal in zowe editor [#129](#)
- Added snackbar notification for directory error [#131](#)
- Removed language server tab in editor [#134](#)
- Explicitly call zss for logout to make sure cookies are known to be invalid [#28](#)
- The following changes have been made to Zlux server framework logging [#174](#):
 - Added English resource files for messages
 - Added code to all error, warning, debug and informational logged outputs
 - Replaced most console.log calls with logger calls
- Support for HTTP-Strict-Transport-Security. Custom headers for static content are now available [#173](#)
- Functionality for controlling application access for individual users has been added [#216](#)
- Out-of-band multi-factor authentication is now supported [#225](#)

Zowe CLI

To leverage the new features and plug-ins available in this version, you must follow the steps in [Migrating to the LTS version](#).

The following new CLI plug-ins are added:

- IBM® z/OS FTP Plug-in for Zowe CLI
- IBM® IMS™ Plug-in for Zowe CLI
- IBM® MQ Plug-in for Zowe CLI
- Secure Credential Store for Zowe CLI

The following new features and enhancements are added in this version:

- **Notable Change:** The `zowe zos-files download ds` and `zowe zos-files download uf` commands no longer put the full content in the response format json (`--rfj`) output. [More information](#).

- **Notable Change:** The `--pass` option is changed to `--password` for all commands and profiles (zosmf, cics, etc...). The aliases `--pw` and `--pass` still function. To update a profile, issue the `zowe profiles update` command and use the new option name `--password`.
- **Notable Change:** You can enter `PROMPT*` as a value for any CLI option to enable interactive prompting. If you wrote scripts in which any option is defined with the exact value `PROMPT*`, the script will not execute properly in this version. For more information, see [Using the prompt for sensitive options](#).
- Zowe CLI was tested and confirmed to run on Unix System Services (USS) on z/OS. For more information, refer to blog [Installing Node.js on the Mainframe](#).

(The IBM Db2 and Secure Credential Store plug-ins for Zowe CLI will *not* run on z/OS due to native code requirements.)

- The `zowe files copy` command was added for copying the contents of a data set or member to another data set or member. [#580](#)
- Zowe CLI now exploits Node.js stream APIs for download and upload of spool files, data sets, and USS files. [\(#331\)](#)
- The following new commands were added for interacting with file systems:
 - `zowe zos-files list fs` [#429](#)
 - `zowe zos-files mount fs` [#431](#)
 - `zowe zos-files unmount fs` [#432](#)
- The following new commands were added for creating USS files and directories:
 - `zowe zos-files create file` [#368](#)
 - `zowe zos-files create dir` [#368](#)

The IBM® CICS® Plug-in is updated with the following functionality:

- **Notable Change:** The plug-in now uses HTTPS by default when connecting to CMCI. The option `--protocol http` was added to let you override the default as needed. [#77](#)
- Define, enable, install, discard, disable, and delete CICS URIMaps. [#53 #49 #48 #51 #50 #52](#)
- Define and delete CICS web services. [#58 #59](#)

- Add and remove CSD Groups to/from CSD Lists [#60](#).

Zowe Explorer

Review the [Zowe Explorer Change Log](#) to learn about the latest features, enhancements, and fixes.

You can install the latest version of the extension from the [Visual Studio Code Marketplace](#).

Watch a video on how to [work with data sets using Zowe Explorer](#).

Bug fixes

The following bugs were fixed:

Zowe App Server

- URL encoding with % sign were always returning with authorization:false with RACF [#27](#)
- Users are no longer able to delete the initial "/" in the address bar for selected files [#379](#)
- The search bar text for datasets has been changed from "Enter a dataset" to "Enter a dataset query". The Address bar text for files has been changed from "Enter a directory" to "Enter an absolute path" [#60](#)

Version 1.8.1 (February 2020)

Bug fixes for Zowe CLI

A bug was fixed where Zowe CLI installation could fail and users could receive the following error message when installing Zowe CLI v1.8.0:

To install the fix, download the new v1.8.1 package from [Zowe.org](#) and retry the installation process.

Version 1.8.0 (February 2020)

New features and enhancements

The following features and enhancements were added.

Installation of Zowe z/OS components

- The installation now just needs two parameters configured: the USS location of the runtime directory and a data set prefix where a SAMPLIB and LOADLIB will be created. The runtime directory permissions are set to 755 and when Zowe is run, no data is written to the runtime directory.
- The way to configure Zowe is changed. Previously, you configured Zowe at installation time with the `zowe-install.yaml` file. This file has been removed and is no longer used in this release.
- A new directory `zowe-instance-dir` has been introduced that contains configuration data used to launch Zowe. This allows more than one Zowe instance to be started from the same Zowe runtime directory. A new file `instance.env` within each `zowe-instance-dir` directory controls which ports are allocated to the Zowe servers as well as location of any dependencies such as Java, z/OSMF or node. No configuration data is specified at install time. The data is only read, validated and used at launch time. The `instance.env` file contains a parameter value `LAUNCH_COMPONENT_GROUPS` that allows you to control which Zowe subsystems to launch, for example you can run the Zowe desktop and not the API Mediation Layer, or vice-versa; you can run just the API Mediation Layer and not the Zowe desktop. The `zowe-instance-dir` directory is also where log files are collected. Static extensions to the API Mediation Layer are recorded in the Zowe instance directory as well as any plug-in extensions to the Zowe desktop. This allows the runtime directory to be fully replaced during PTF upgrades or moving to later Zowe releases while preserving configuration data and extension definitions that are held in the instance directory.
- A new directory `keystore-directory` has been introduced outside of the Zowe runtime directory which is where the Zowe certificate is held, as well as the truststore for public certificates from z/OS services that Zowe communicates to (such as z/OSMF). A keystore directory can be shared between multiple Zowe instances and across multiple Zowe runtimes.
- All configuration of z/OS security that was done by Unix shell scripts during installation and configuration has been removed. A JCL member `ZWESECUR` is provided that contains all of the JCL needed to configure permissions, user IDs and groups, and other steps to prepare and configure a z/OS environment to successfully run Zowe. Code is included for RACF, Top Secret, and ACF/2.
- The Zowe cross memory server installation script `zowe-install-apf-server.sh` is removed. In this release, the steps for configuring z/OS security are included in the `ZWESECUR` JCL member.
- Previously, Zowe runs its two started tasks under the user ID of `IZUSVR` and admin of `IZUADMIN`. These belong to z/OSMF and are no longer used in this release. Instead, Zowe includes two new user IDs of `ZWESVUSR` (for the main Zowe started task), `ZWESIUSR` (for the cross memory server), and `ZWEADMIN` as a group. These user IDs are defaults and different ones can be used depending on site preferences.
- Previously, the main Zowe started task is called `ZOWESVR`. Now it is called `ZWESVSTC`.
- Previously, the cross memory started task is called `ZWESIS01`. Now it is called `ZWESISTC`.

- The script `zowe-verify.sh` is no longer included with Zowe. Now the verification is done at launch time and dependent on the launch configuration parameters. It is no longer done with a generic script function that `zowe-verify.sh` used to provide.

For more information about how to install Zowe z/OS components, see [Installation roadmap](#).

API Mediation Layer

- The API Catalog backend has been modified to support the OpenAPI 3.0 version. The API Catalog now supports the display of API documentation in the OpenAPI 3.0 format.
- A new Eureka metadata definition has been developed to enable service registration that does not require using existing pre-prepared enablers. Both new and old metadata versions are supported by the Discovery Service. Corresponding documentation to onboard a service with the Zowe API ML without an onboarding enabler has also been refactored.
- The plain Java enabler has been redesigned for simple and straight-forward API service configuration. Configuration parameters have been refactored to remove duplicates and unused parameters, and improve consistency with other parameters. Documentation to Onboard a REST API service with the Plain Java Enabler (PJE) has also been refactored.

Zowe App Server

- The app server now issues a message indicating it is ready, how many plug-ins loaded, and where it can be accessed from [#355](#)
- Restructured the App server directories to separate writable configuration items from read-only install content [#911](#) [#627](#) [#87](#) [#43](#)
- Move install-app script to instance directory bin folder for ease of use [#966](#)
- Access control for app visibility [216](#)
- The following features and enhancements were made in the default apps:
 - UI changes for write support for datasets in editor [#340](#)
 - Support for QSAM and VSAM deletion in the ZSS dataset REST API [#339](#)
 - Editor: Dataset deletion capability [[#229](#)][[#337](#)]
 - Editor: File deletion UI changes [#338](#)
 - Editor fix: When saving a new file use the opened directory in the dialog [#233](#)
 - Editor fix: Disable text area for datasets in the absence of write ability [#342](#)
 - Editor fix: When saving a new file use the opened directory in the dialog [#233](#)

Zowe CLI

- The Zowe CLI REST API now supports the following capabilities for managing data sets:
 - Rename sequential and partitioned data sets. [#571](#)
 - Migrate data sets. [#558](#)
 - Copy data sets to another data set and copy members to another member. [#578](#)
- The Zowe CLI REST API now supports HTTP ETags in response data. The ETag mechanism allows client applications to cache data more efficiently. ETags can also prevent simultaneous, conflicting updates to a resource. [#598](#)

Zowe Explorer

Review the [Zowe Explorer Change Log](#) to learn about the latest features, enhancements, and fixes.

You can install the latest version of the extension from the [Visual Studio Code Marketplace](#).

Check the new "Getting Started with Zowe Explorer" video to learn how to install and get started with the extension. For more information, see [Zowe Explorer Extension for VSCode](#).

Bug fixes

The following bugs were fixed.

Zowe App Server

- Use of environment variables (_TAG_REDIR_XXX) required to run Zowe with node v12 [#333](#)
- `install-app.sh` script would not work without first server run, improper permissions [#373](#)

Zowe CLI

- Fixed an issue where `zowe zos-jobs submit stdin` command returned an error when handling data from standard in. [#601](#)
- Updated dependencies to address potential vulnerabilities. Most notably, Yargs is upgraded from v8.0.2 to v15.0.2. [#333](#)

Version 1.7.1 (December 2019)

New features and enhancements

The following features and enhancements were added.

Zowe App Server

- A backup routine for when a non-administrator tries to access the API. Instead of executing privileged commands and failing, it will execute a command to get their profile, and return only the information in their scope. This is a feature that most people won't need, since you'd ideally want to be an administrator if you were using this API, but the functionality is there. ([#114](#))
- The ability to retrieve profiles only by prefix. This can be done by looking for a profile with a "." at the end. This will act as a wildcard which extracts everything matching that prefix. ([#114](#))

Zowe SMP/E installation

The pre-release of the Zowe SMP/E build is updated to be based on Zowe Version 1.7.1.

Bug fixes

The following bugs were fixed.

Zowe App Server

- Fixed a bug where the end of an acid is cut off when getting the access list of a group, resulting in invalid output in the response. ([#114](#))
- Fixed a bug where all of the different administrator suffixes weren't defined, so it was incorrectly returning administrators. ([#114](#))

Version 1.7.0 (November 2019)

New features and enhancements

The following features and enhancements were added.

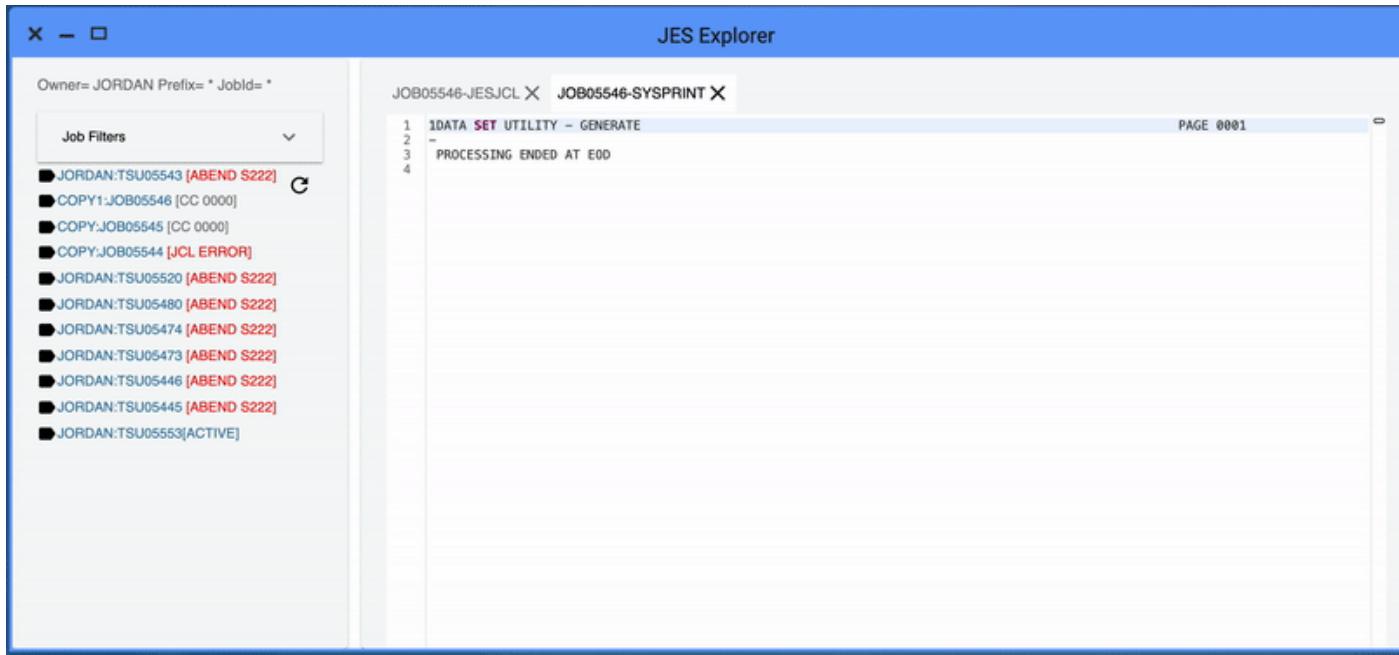
API Mediation Layer

- Cleanup Gateway dependency logs ([#413](#))
- Cleanup Gateway - our code ([#417](#))
- Cleanup Discovery Service dependency logs ([#403](#))
- Cleanup Discovery Service - our code ([#407](#))
- External option to activate DEBUG mode for APIML ([#410](#))

Zowe App Server

- Introduced the "SJ" feature to the JES Explorer application ([#282](#))

You can now right-click a job label and click "Get JCL" to retrieve the JCL used to submit the job. This JCL can then be edited and resubmitted.



- File Explorer now offers a right click Delete option for files and folders ([#43](#))
- Prevented creation/deletion of files and folders queued for deletion. ([#48](#))
- Updated back-end API to give more accurate delete responses. ([#93](#))
- IFrame adapter: added support for plugin definition, logger, and launch metadata. ([#174](#))
- Iframe app-to-app communication support ([#174](#))
- Removed unnecessary warning suppression ([#23](#))
- Dispatcher always sends message, even when context doesn't exist ([#174](#))
- Support constructor injectables via Iframe adapter ([#174](#))
- Browser tab for the desktop now includes opened app name. ([#175](#))
- File Explorer now offers a right click file and folder Properties menu. ([#180](#))
- File Explorer now offers a right click dataset Properties menu. ([#49](#))

- Made it possible to specify config properties via command line arguments for the App server. ([#81](#))
- Allow override of configuration attributes using a -D argument syntax. ([#154](#))
- Allow specifying environment variables that can be interpreted as JSON structures. ([#156](#))

Zowe Explorer (Extension for VSCode)

- The name of the extension was changed from "VSCode Extension for Zowe" to "Zowe Explorer".
- The VSCode Extension for Zowe contains various changes in this release. For more information, see the [VSCode Change Log](#).

Bug fixes

The following bugs were fixed.

API Mediation Layer

Fixed a typo in Gateway startup script. ([#427](#))

Zowe App Server

Fixed notification click, time stamp, inconsistent notification manager pop up clicks, empty notification bubbles, and safari issue. ([#171](#))

Zowe CLI

This version of Zowe CLI contains various bug fixes that address vulnerabilities.

Version 1.6.0 (October 2019)

No changes were made to API ML or Zowe CLI in this release.

What's new in the Zowe App Server

The following features and enhancements are added:

- Added two NodeJS issues to the App Framework Troubleshooting section. [#786](#)
- Added a REST API for new core dataservices to administer the servers and plugins. [#82](#)
- Added pass through express router ws patcher in case plug-ins need it. [#152, #149](#)

- Updated security plugins to manage proxied headers so that unnecessary things are not put into the browser.[#152](#), [#26](#)
- Clear cookie on complete logout.[#152](#)

What's new in Zowe CLI

The following enhancement was added:

- The `--wait-for-output` and the `--wait-for-active` options were added. You can append these options to a `zowe zos-jobs submit` command to either wait for the job to be active, or wait for the job to complete and enter OUTPUT status. If you do not specify `--vasc`, you can use these options to check job return codes without issuing `zowe zos-jobs view job-status-by-jobid <jobid>`.

What's new in the Visual Studio Code (VSC) Extension for Zowe

The Visual Studio Code (VSC) Extension for Zowe lets you interact with data sets and USS files from a convenient graphical interface. Review the [Change Log](#) to learn about the latest improvements to the extension.

You can [download the latest version](#) from the VSC Marketplace.

Version 1.5.0 (September 2019)

What's new in API Mediation Layer

The following features and enhancements are added:

- The Discovery Service UI now enables the user to log in using mainframe credentials or by providing a valid client certificate.
- API Catalog REST endpoints now accept basic authentication by requiring the user to provide a username and password.

The following bugs are fixed:

- A defect has been resolved where previously an authentication message was thrown in the service detail page in the API Catalog when the swagger JSON document link was clicked. The message requires the user to provide mainframe credentials but did not link to an option to authenticate. Now, a link is included to provide the user with the option to authenticate.

What's new in the Zowe App Server

The following features and enhancements are added:

- Adds dynamic logging functionality for plugins ([#60](#), [#63](#))
- Top Secret updates to the security lookup API ([#71](#), [#72](#), [#74](#))
- Accept basic auth header as an option for login ([#80](#))
- JSON parsing enhancements for UTF8, and printing to buffer ([#67](#))
- Optimization, memory bugfix and improved tracing for authentication ([#72](#))
- Performance optimization for app thumbnail snapshots: Fixed a bug causing slowdown relative to number of apps open ([#131](#))
- Translations: Added missing language translations about session lifecycle ([#137](#))
- Logger reorganized for Zowe-wide log format unification. Includes i18n-able message ID support & new info. See [#90](#) ([#17](#), [#119](#), [#116](#), [#142](#), [#35](#), [#19](#), [#132](#), [#146](#), [#126](#), [#139](#), [#67](#), [#133](#), [#21](#))
- Establish rules & recommendations for conformance ([#142](#))
- Launchbar menu of apps now has same context menu properties as pinned apps ([#140](#))
- Properties App now shows the ID of the chosen plugin ([#140](#))
- Added group permission for plugin access when installing via install script ([#125](#))
- Updated URIBroker include new parameter for searching datasets with included trailing qualifiers ([#34](#), [#138](#))
- App2App communication now allows you to target a specific app instance, as well as to request minimization or maximization ([#38](#), [#148](#))
- Configuration Dataservice now can load plugin defaults from the plugin's own folder ([#129](#))
- Configuration Dataservice can now support GET like HEAD ([#140](#))
- Configuration Dataservice can now utilize binaries as opposed to JSON. This mode does not process the objects, just stores & retrieves. ([#130](#))
- Added a notification menu, popup & API where messages can be sent by administrators to individual or all end users ([#36](#), [#144](#))
- Doc: Configuration Dataservice Swagger document updated for new features ([#136](#))
- Desktop now supports loading a custom wallpaper, and the launchbar & maximized window style has been changed to improve screen real estate ([#151](#))
- The App Server configuration and log verbosity can now be viewed and updated on-the-fly via a REST API ([#66](#), [#128](#))
- The App Server environment parameters and log output can now be viewed via a REST API ([#66](#), [#128](#))

- The App Server can now have Application plugins added, removed, and upgraded on-the-fly via a REST API ([#137](#), [#69](#))
- A dataservice can now import another import dataservice, as long as this chain eventually resolves to a non-import dataservice ([#139](#))
- You can now open any Zowe App in its own browser tab by right clicking its icon and choosing "Open in new browser window" ([#149](#), [#150](#))
- Icons improved for datasets that are migrated/archived ([#30](#))
- Support App2App to open a given dataset ([#87](#), [#35](#))
- Navigate the editor menu bar via keyboard ([#85](#))
- Add keyboard shortcuts to open and close tabs ([#81](#))
- Add loading indicator for dataset loading ([#34](#))
- Compress the terminals with gzip for improved initial load time, same as was done with the editor previously ([#22](#), [#23](#))
- Made the following enhancements to the JES Explorer App
 - Add ability to open and view multiple Spool files at once ([#99](#))
 - Migrate from V0 to V1 of Material UI ([#98](#))
 - Migrate from V15 to V16 of React ([#98](#))

The following bugs are fixed:

- New directories/files from Unix file API would have no permissions ([#75](#))
- Properties App can now be reused when clicking property of a second app ([#140](#))
- Logout did not clear dispatcher App instance tracking ([#32](#))
- Iframe Apps were not gaining mouse focus correctly ([#37](#), [#145](#))
- Remove placeholder swagger from swagger response when plugin-provided swagger is found ([#139](#))
- ZSS Dataservices could fail due to incorrect impersonation environment variable setting (_BPX_SHAREAS) ([#68](#))
- Restore focus of text on window restore ([#84](#))
- Reposition menu from menu bar on Edge/Firefox ([#82](#))
- Could not open the SSH terminal in single window mode ([#21](#))

What's new in Zowe CLI and Plug-ins

The following commands and enhancements are added:

- You can append `--help-web` to launch interactive command help in your Web browser. For more information, see [Interactive Web Help. \(#238\)](#)

Zowe SMP/E Alpha (August 2019)

A pre-release of the Zowe SMP/E build is now available. This alpha release is based on Zowe Version 1.4.0. Do not use this alpha release in production environment.

- To obtain the SMP/E build, go to the [Zowe Download](#) website.
- For more information, see [Installing Zowe SMP/E Alpha](#).

Version 1.4.0 (August 2019)

What's new in API Mediation Layer

This release of Zowe API ML contains the following improvements:

- JWT token configuration
 - RS256 is used as a token encryption algorithm
 - JWT secret string is generated at the time of installation and exported as a `.pem` file for use by other services
 - JWT secret string is stored in a key store in PKCS 11 format under "jwtsecret" name
- SonarQube problems fixed
 - Various fixes from SonarQube scan
- API Mediation Layer log format aligned with other Zowe services:
- Added an NPM command to register certificates on Windows. The following command installs the certificate to trusted root certification authorities:
- Cookie persistence changed
 - Changed the API Mediation Layer cookie from persistent to session. The cookie gets cleared between browser sessions.
- Fixed high CPU usage occurrence replicated in Broadcom ([#282](#))

- Changed configuration of LatencyUtils to decrease idle CPU consumption by API ML services
- API Mediation layer now builds using OpenJDK with OpenJ9 JVM

What's new in the Zowe App Server

Made the following fixes and enhancements:

- Added the ability for the App Server Framework to defer to managers for dataservices that are not written in NodeJS or C. The first implementation is a manager of Java servlet type dataservices, where the App Server manages Tomcat instances when Tomcat is present. ([#158](#))
- Added a tomcat xml configuration file with substitutions for values (ports, keys, certificates) necessary for the App Server to manage one or more instances of Tomcat for hosting servlet dataservices. Also added a new section to the zluxserver.json file to describe dataservice providers such as the aforementioned Tomcat Java Servlet one. ([#49](#))
- Added Swagger API documentation support. Application developers can include a Swagger 2.0 JSON or YAML file in the app's /doc/swagger directory for each REST data service. Each file must have the same name as the data service. Developers can then reference the files at runtime using a new app route: /ZLUX/plugins/PLUGINID/catalogs/swagger. They can reference individual services at: /ZLUX/plugins/PLUGINID/catalogs/swagger/SERVICENAME. If swagger documents are not present, the server will use contextual knowledge to show some default values. ([#159](#))
- The following new REST and cross-memory services have been added ([#32](#)):
 - Extract RACF user profiles
 - Define/delete/permit general RACF resource profiles (limited to a single class)
 - Add/remove RACF groups
 - Connect users to RACF groups (for a limited set of group prefixes)
 - Check RACF user access levels (limited to a single class)
- Fixed multiple issues in the File Editor App. ([#88](#))
- Fixed multiple ZSS file and dataset API issues ([#49](#) [#42](#) [#40](#) [#44](#) [#45](#))
- Remove several CSS styles from the Desktop to prevent bleed-in of styles to Apps ([#117](#))
- Fixed incorrect count of open Apps upon logging in more than once per browser session ([#123](#))
Add OMVS information API to uribroker ([#116](#))
- Enhanced auth plugin structure for application framework that lists auth capabilities ([#118](#) [#14](#) [#19](#))
- Improved searching for node libraries for dataservices within an plugin ([#114](#))
- Editor & File Explorer Widget Changes
 - Unix directory listing now starts in the user's home directory ([#16](#))

- JCL syntax coloring revision ([#73](#))
- Cursor, scroll position and text selection is now kept while switching tabs in editor ([#71](#))
- Editor now scrolls tab bar to newest tab when opening, and tab scrolling improved when closing tabs ([#69](#))
- Tab name, tooltip, and scroll fixes ([#55](#) [#60](#) [#63](#))
- Change in double and single click behavior of file explorer widget ([#21](#))
- Fix to show language menu on new file ([#62](#))
- Fix to keep language menu within the bounds of app window ([#59](#))
- Fix to the delete file prompt ([#61](#))
- Fix to allow closing of multiple editor instances ([#22](#))
- Fix to query datasets correctly by making queries uppercase ([#65](#))
- Fixed issue where the cascading position of new windows were wrong when that application was maximized. ([#102](#))
- Fixed issue where the file tabs in File Editor app were vertically scrollable, and where the close button would not be accessible for long file names. ([#170](#))
- Updated the package lock files in all repositories to fix vulnerable dependencies. ([#163](#))
- Fixed an issue where the Desktop used the roboto-latin-regular font for all text, which would not display well with non-latin languages. Now the fallback font is sans-serif. ([#118](#))

What's new in Zowe CLI and Plug-ins

You can now explore the Zowe CLI command help in an interactive online format. See [Zowe CLI Web Help](#).

The following new commands and enhancements are added:

- The [VSCode Extension for Zowe](#) now supports manipulation of USS files. ([#32](#))
- You can now archive z/OS workflows using a wildcard. ([#435](#))
- The z/OS Workflows functionality is now exported to an API. Developers can leverage the exported APIs to create applications and scripts without going through the CLI layer. ([#482](#))
- The CLI now exploits all "z/OS data set and file REST interface" options that are provided in z/OSMF v2.3. ([#491](#))

The following bugs are fixed:

- Fixed an issue where examples for `zowe files list uss-files` were slightly incorrect. ([#440](#))
- Improved error message for `zowe db2 call procedure` command. ([#22](#))

Version 1.3.0 (June 2019)

What's new in API Mediation Layer

This release of Zowe API ML contains the following user experience improvements:

- Added authentication endpoints (/login, /query) to the API Gateway
- Added the Gateway API Swagger document ([#305](#))
 - Fixed the bug that causes JSON response to set incorrectly when unauthenticated
 - Fixed error messages shown when a home page cannot be modified
- Added a new e2e test for GW, and update the detail service tile ([#309](#))
- Removed a dependency of integration-enabler-java on the gateway-common ([#302](#))
- Removed access to the Discovery service UI with basic authentication ([#313](#))
- Fixed the issue with the connection logic on headers to pass in the websocket ([#275](#))
- Fixed the bug 264: Bypass the API Gateway when the server returns 302 ([#276](#))
- Fixed the issue that causes the API ML Services display as UP, and makes the API doc available in the Catalog regardless whether the API ML Services stop ([#287](#))
- Fixed the issue that prevents the API Catalog to load under zLux 9 ([#314](#))

What's new in the Zowe App Server

Made the following fixes and enhancements:

- Added internationalization to the Angular and React sample applications. ([#133](#))
- Made the following enhancements to the ZSS server:
 - Added support for Zowe on z/OS version 2.4. ([#15](#))
 - Updated documentation for query parameter to file API. ([#48](#))
- Made the following enhancements to security:
 - App Server session cookie is now a browser session cookie rather than having an expiration date. Expiration is now tracked on the server side. ([#132](#), [#97](#), [#81](#))
 - Added a "mode=base64" option to the unixfile API. ([#127](#))
- Added a port to the cookie name to differentiate multiple servers on same domain. ([#95](#))
- Made the following fixes and enhancements to the Code Editor application:
 - Added a menu framework to provide options specific to the current file/data set type. ([#131](#))
 - Added ISPF-like syntax highlighting for JCL. ([#48](#))
 - Fixed an issue by notifying users if the editor cannot open a file or data set. ([#148](#))

- Fixed an issue with event behavior when a tab is closed. ([#135](#))
 - Fixed an issue with not showing the content of files in Chrome and Safari. ([#100](#))
 - Fixed an issue with files shown without alphabetical sorting. ([#85](#))
- Made the following fixes and enhancements to the TN3270 application ([#96](#)):
 - Fixed an issue where the application could not be configured to default to a TLS connection.
 - Fixed an issue where it could not handle a TN3270 connection, only TN3270E. Improved preference saving. Administrators can now store default values for terminal mod type, codepage, and screen dimensions.
- Made the following fixes and enhancements for App2App for IFrames ([#24](#), [#107](#)):
 - Fixed an issue with an exception when handling App2App communication with IFrames.
 - Added experimental support for App2App communication with an IFrame application as destination.
- Made the following enhancements to support TopSecret:
 - Added a user-profiles endpoint. ([#113](#))
 - Added an endpoint extraction for groups. ([#129](#))
- Fixed an issue with app names not being internationalized when translations were present. ([#85](#))
- Fixed Russian language errors in translation files. ([#100](#))
- Fixed several issues with using the Application Server as a proxy. ([#93](#))
- Fixed an issue with the App Server throwing exceptions when authorization plugins were installed but not requested. ([#94](#))
- Fixed an issue with ZSS consuming excessive CPU during download. ([#147](#))
- Fixed documentation issue by replacing "zLUX" with "Zowe Application Framework" and "MVD" with "Zowe Desktop." ([#214](#))
- Fixed an issue with an incorrect translation for word "Japanese" in Japanese. ([#108](#))

What's new in Zowe CLI and Plug-ins

The following new commands and enhancements are added:

- Return a list of archived z/OSMF workflows with the `zowe zos-workflows list arw` command. ([#391](#))
- Return a list of systems that are defined to a z/OSMF instance with the `zowe zosmf list systems` command. ([#348](#))
- The `zowe uss issue ssh` command now returns the exit code of the shell command that you issued. ([#359](#))

- The `zowe files upload dtu` command now supports the metadata file named `.zosattributes`. (#366)

The following bugs are fixed:

- Fixed an issue where `zowe workflow ls aw` commands with the `--wn` option failed if there was a space in the workflow name. (#356)
- Fixed an issue where `zowe zowe-files delete uss` command could fail when resource URL includes a leading forward-slash. (#343).

Version 1.2.0 (May 2019)

Version 1.2.0 contains the following changes since Version 1.1.0.

What's new in the Zowe installer

- Made the following installer improvements:
 - Check whether ICSF is configured before checking Node version to avoid runaway CPU.
 - Warn if the host name that is determined by the installer is not a valid IP address.
 - Fixed a bug where a numeric value is specified in ZOWE_HOST_NAME causing errors generating the Zowe certificate.
- Made the following improvements to the `zowe-check-prereqs.sh` script:
 - Improvements for checking and validating the telnet and ssh port required by the Zowe Desktop applications.

What's new in API Mediation Layer

This release of Zowe API ML contains the following user experience improvements:

- Prevented the Swagger UI container on the service detail page from spilling.
- Added a check for the availability of the z/OSMF URL contained in the configuration. z/OSMF is used to verify users logging into the Catalog.
- Made *PageNotFound* error visible only in the debug log level.
- Added zD&T-compatible ciphers and the TLS protocol restricted to 1.2.
- Introduced support for VSCode development.
- Introduced a common cipher configuration property.
- Fixed URL transformation defects.

- Fixed reporting that the Catalog is down when it is started before the Discovery Service.
- Removed the bean overriding error message from the log.
- Fixed the state manipulation mechanism in the Catalog. As a result, no restoring of the application state is performed.
- Fixed the Catalog routing mechanism for a users who is already logged in so that the user is not prompted to log in again.
- A timeout has been set for Catalog login when z/OSMF is not responding.
- A tile change in the Catalog is now propagated to the UI.
- Fixed a problem with an incorrect service homepage link in the Catalog.
- The Catalog Login button has been disabled when the login request is in progress.

What's new in the Zowe App Server

- Improved security by adding support for RBAC (Role Based Access Control) to enable Zowe to determine whether a user is authorized to access a dataservice.
- Added Zowe Desktop settings feature for specifying the Zowe desktop language.
- Added German language files.
- Fixed a bug by adding missing language files.
- Enabled faster load times by adding support for serving the Zowe Application Framework core components, such as the Desktop, as compressed files in gzip format.
- Added support for application plug-ins to serve static content, such as HTML, JavaScript, and images, to browsers in gzip and brotli compressed files.
- Fixed a Code Editor bug by separating browsing of files and data sets.

What's new in Zowe CLI and Plug-ins

The Zowe CLI core component contains the following improvements and fixes:

- The `zos-uss` command group is added to the core CLI. The commands let you issue Unix System Services shell commands by establishing an SSH connection to an SSH server. For more information, see .
- The zowe `zos-workflows` command group now contains the following `active-workflow-details` options:
 - `--steps-summary-only | --sso (boolean)` : An optional parameter that lets you list (only) the steps summary.

- `--skip-workflow-summary | --sws (boolean)`: An optional parameter that lets you skip the default workflow summary.
- Zowe CLI was updated to correct an issue where the `zowe zos-workflows start` command ignored the `-- workflow-name` argument.
- Updated and clarified the description the `-- overwrite` option for the `zowe zos-workflows create workflow-from-data-set` command and the `Zowe zos-workflows create workflow-from-uss-file` command.
- The [CLI Reference Guide](#) is featured on the Zowe Docs home page. The document is a comprehensive guide to commands and options in Zowe CLI.
- You can now click the links on the Welcome to Zowe help section and open the URL in a browser window. Note that the shell application must support the capability to display and click hyperlinks.

What's new in Zowe USS API

Made the following enhancements:

- Chtag detection and ASCII/EBDCIC conversion on GET & PUT requests. For details, see [this issue](#).
- New optional header on GET Unix file content request to force conversion from ebcdic to ascii. For details, see [this issue](#).
- New response header on GET Unix file content requests: E-Tag for overwrite detection and validation. For details, see [this issue](#).
- Reintroduced PUT (update) Unix file content endpoint. For details, see [this issue](#).
- Reintroduced DELETE Unix file content endpoint. For details, see [this issue](#).
- Reintroduced POST (create) Unix file or directory endpoint. For details, see [this issue](#).
- Fixed a problem with incorrect return error when the user requests to view contents of a USS folder they do not have permission to. Now it returns a 403 (Forbidden) error. For details, see [this issue](#).

Version 1.1.0 (April 2019)

Version 1.1.0 contains the following changes since the last 1.0.x version.

What's new in Zowe system requirements

z/OSMF Lite is now available for non-production use such as development, proof-of-concept, demo and so on. It simplifies the setup of z/OSMF with only a minimal amount of z/OS customization, but provides key functions that are required. For more information, see [Configuring z/OSMF Lite \(for non-production use\)](#).

What's new in the Zowe App Server

- Made the following user experience improvements:
 - Enabled the Desktop to react to session expiration information from the Zowe Application Server. If a user is active the Desktop renews their session before it expires. If a user appears inactive they are prompted and can click to renew the session. If they don't click, they are logged out with a session expired message.
 - Added the ability to programmatically dismiss popups created with the "zlux-widgets" popup manager.
- Made the following security improvements:
 - Encoded URIs shown in the App Server 404 handler, which prevents some browsers from loading malicious scripts.
 - Documented and support configuring HTTPS on ZSS.
 - For ZSS API callers, added HTTP response headers to instruct clients not to cache HTTPS responses from potentially sensitive APIs.
- Improved the Zowe Editor App by adding app2app communication support that allows the application to open requested directories, dataset listings, and files.
- Improved the Zowe App API by allowing subscription to close events on viewports instead of windows, which allows applications to better support Single App Mode.
- Fixed a bug that generated an extraneous RACF audit message when you started ZSS.
- Fixed a bug that would sometimes move application windows when you attempted to resized them.
- Fixed a bug in the "Getting started with the ZOWE WebUi" tutorial documentation.
- Fixed a bug that caused applications that made ZSS service requests to fail with an HTTP 401 error because of dropped session cookies.

What's new in the Zowe CLI and Plug-ins

This release of Zowe CLI contains the following new and improved capabilities:

- Added APIs to allow the definition of workflows

- Added the option `max-concurrent-requests` to the `zowe zos-files upload dir-to-uss` command
- Added the option `overwrite` to the `zowe zos-workflows create` commands
- Added the option `workflow-name` to the `zowe zos-workflows` commands
- Added the following commands along with their APIs:
 - `zowe zos-workflows archive active-workflow`
 - `zowe zos-workflows create workflow-from-data-set`
 - `zowe zos-workflows create workflow-from-uss-file`
 - `zowe zos-workflows delete active-workflow`
 - `zowe zos-files list uss-files`

This release of the Plug-in for IBM DB2 Database contains the following new and improved capabilities:

- Implemented command line precedence, which lets users issue commands without the need of a DB2 profile.
- The DB2 plug-in can now be influenced by the `ZOWE_OPT_` environment variables.

What's new in API Mediation Layer

- Made the following user experience improvements:
 - Documented the procedure for changing the log level of individual code components in *Troubleshooting API ML*.
 - Documented a known issue when the API ML stops accepting connections after z/OS TCP/IP is recycled in the *Troubleshooting API ML*.

Version 1.0.1 (March 2019)

Version 1.0.1 contains the following changes since the last version.

What's new in Zowe installation on z/OS

During product operation of the Zowe Cross Memory Server which was introduced in V1.0.0, the z/OSMF user ID IZUSVR or its equivalent must have UPDATE access to the BPX.SERVER and BPX.DAEMON FACILITY classes. The install script will do this automatically if the installing user has enough authority, or provide the commands to be issued manually if not. For more information, see [Installing the Zowe Cross Memory Server on z/OS](#)

What's new in the Zowe App Server

- Made the following improvements to security:
 - Removed the insecure SHA1 cipher from the Zowe App Server's supported ciphers list.
 - Added instructions to REST APIs to not cache potentially sensitive response contents.
 - Set secure attributes to desktop and z/OSMF session cookies.
- Fixed a bug that caused the configuration data service to mishandle PUT operations with bodies that were not JSON.
- Fixed a bug that prevented IFrame applications from being selected by clicking on their contents.
- Fixed various bugs in the File Explorer and updated it to use newer API changes.
- Fixed a bug in which App2App Communication Actions could be duplicated upon logging in a second time on the same desktop.

What's new in Zowe CLI

- Create and Manage z/OSMF Workflows using the new `zos-workflows` command group. For more information, see [Zowe CLI command groups](#).
- Use the `@lts-incremental` tag when you install and update Zowe CLI core or plug-ins. The tag ensures that you don't consume breaking changes that affect your existing scripts. Installation procedures are updated to reflect this change.
- A [CLI quick start guide](#) is now available for users who are familiar with command-line tools and want to get up and running quickly.
- IBM CICS Plug-in for Zowe CLI was updated to support communication over HTTPS. Users can enable https by specifying `--protocol https` when creating a profile or issuing a command. For backwards compatibility, HTTP remains the default protocol.

What's new in the Zowe REST APIs

Introduced new Unix files APIs that reside in the renamed API catalog tile `z/OS Datasets and Unix files service` (previously named `z/OS Datasets service`). You can use these APIs to:

- List the children of a Unix directory
- Get the contents of a Unix file

What's changed

- **Zowe explorer apps**

- JES Explorer: Enhanced Info/Error messages to better help users diagnose problems.
- MVS Explorer: Fixed an issue where Info/Error messages were not displayed when loading a Dataset/Members contents.

Version 1.0.0 (February 2019)

Version 1.0.0 contains the following changes since the Open Beta release.

What's new in API Mediation Layer

- HTTPs is now supported on all Java enablers for onboarding API microservices with the API ML.
- SSO authentication using z/OSMF has been implemented for the API Catalog login. Mainframe credentials are required for access.

What's new in Zowe CLI

- **Breaking change to Zowe CLI:** The `--pass` command option is changed to `--password` for all core Zowe CLI commands for clarity and to be consistent with plug-ins. If you have zosmf profiles that you created prior to January 11, 2019, you must recreate them to use the `--password` option. The aliases `--pw` and `--pass` still function when you issue commands as they did prior to this breaking change. You do not need to modify scripts that use `--pass`.
- The `@next` npm tag used to install Zowe CLI is deprecated. Use the `@latest` npm tag to install the product with the online registry method.

What's new in the Zowe Desktop

- You can now obtain information about an application by right-clicking on an application icon and then clicking **Properties**.
- To view version information for the desktop, click the avatar in the lower right corner of the desktop.
- Additional information was added for the Workflow application.
- The titlebar of the active window is now colored to give an at-a-glance indication of which window is in the foreground.

- Window titlebar maximize button now changes style to indicate whether a window is maximized.
- Windows now have a slight border around them to help see boundaries and determine which window is active.
- Multiple instances of the same application can be opened and tracked from the launchbar. To open multiple instances, right-click and choose **Open New**. Once multiple instances are open, you can click the application icon to select which application to bring to the foreground. The number of orbs below the application icon relates to the number of instances of the application that is open.
- Desktop framework logging trimmed and formalized to the Zowe App Logger. For more information, see <https://github.com/zowe/zlux/wiki/Logging>.
- The UriBroker was updated to support dataservice versioning and UNIX file API updates.
- Removed error messages about missing `components.js` by making this optional component explicitly declared within an application. By using the property "webContent.hasComponents = true/false".
- Set the maximum username and password length for login to 100 characters each.
- Applications can now list `webContent.framework = "angular"` as an alias for "`angular2`".
- Fixed a bug where the desktop might not load on high latency networks.

What's new in the Zowe App Server

- HTTP support was disabled in favor of HTTPS-only hosting.
- The server can be configured to bind to specific IPs or to hostnames. Previously, the server would listen on all interfaces. For more information, see <https://github.com/zowe/zlux-app-server/pull/30>.
- The core logger prefixes for the Zowe App Server were changed from "`_unp`" to "`_zsf`".
- Dataservices are now versioned, and dataservices can depend on specific versions of other dataservices. A plug-in can include more than one version of a dataservice for compatibility. For more information, see <https://github.com/zowe/zlux/wiki/ZLUX-Dataservices>.

- Support to communicate with the API Mediation Layer with the use of keys and certificates was added.
- Trimmed and corrected error messages regarding unconfigured proxies for clarity and understanding. For more information, see <https://github.com/zowe/zlux-server-framework/pull/33>.
- Fixed the `nodeCluster.sh` script to have its logging and environment variable behavior consistent with `nodeServer.sh`.
- Removed the "swaggerui" plug-in in favor of the API Catalog.
- Bugfix for `/plugins` API to not show the installation location of the plug-in.
- Bugfix to print a warning if the server finds two plug-ins with the same name.
- Added the ability to conditionally add HTTP headers for secure services to instruct the browser not to cache the responses. For more information, see <https://github.com/zowe/zlux-server-framework/issues/36>.
- Added a startup check to confirm that ZSS is running as a prerequisite of the Zowe App Server.
- Bugfix for sending HTTP 404 response when content is missing, instead of a request hanging.
- Added tracing of login, logout, and HTTP routing so that administrators can track access.

What's changed

- Previously, APIs for z/OS Jobs services and z/OS Data Set services are provided sing an IBM WebSphere Liberty web application server. In this release, they are provided using a Tomcat web application server. You can view the associated API documentation corresponding to the z/OS services through the API Catalog.
- References to `zlux-example-server` were changed to `zlux-app-server` and references to `zlux-proxy-server` were changed to `zlux-server-framework`.

Known issues

Paste operations from the Zowe Desktop TN3270 and VT Terminal applications

TN3270 App - If you are using Firefox, the option to use Ctrl+V to paste is not available. Instead, press Shift + right-click to access the paste option through the context menu.

Pressing Ctrl+V will perform paste for the TN3270 App on other browsers.

VT Terminal App - In the VT Terminal App, Ctrl+V will not perform a paste operation for any browser.

Note: In both terminals, press Shift + right-click to access copy and paste options through the context menu.

z/OS Subsystems App - The z/OS Subsystems application is being removed temporarily for the 1.0 release. The reason is that as the ZSS has transitioned from closed to open source some APIs needed to be re-worked and are not complete yet. Look for the return of the application in a future update.

Getting started with Zowe

If you just started exploring Zowe, follow the path in this topic.

Learning about Zowe

Learn what is Zowe, how Zowe works and what it can do for you.

- [Introduction video](#)

Begin your journey with Zowe through this introductory video which is designed to give you an overview of Zowe's various components and features and get you familiar with them quickly.

- [Zowe overview](#)

Refer to the overview topics to walk through basic concepts of Zowe, its components and benefits.

- [Zowe architecture](#)

Review the Zowe architecture to understand how Zowe works.

- [Zowe FAQs](#)

If you have a question, it's a good idea to try the FAQ, which answers the most commonly asked questions about Zowe.

- [Release notes](#)

Check out what's new with Zowe.

- [Zowe learning resources](#)

Find a list of helpful Zowe resources on the web, including blogs, videos, webinars, courses, trials, and so on.

Install and use

Now you know that Zowe consists of several components. You can determine which Zowe components to install and use. Follow the component-specific information roadmaps to install, use, and troubleshoot Zowe components. You can also get information about how to contribute to each specific component.

Components to install and use	Information roadmaps
Zowe Application Framework, Zowe API Mediation Layer (these are installed as z/OS server components)	Information roadmap for Zowe Application Framework , Information roadmap for Zowe API Mediation Layer
Zowe CLI	Information roadmap for Zowe Explorer
Zowe SDKs (under development)	Information roadmap for Zowe client SDKs
Zowe Mobile (Incubator)	Information roadmap for Zowe Mobile

Getting involved

There are many ways to contribute to Zowe, and we welcome contributions! Read the [contributor's guide](#) to get started on the code and guidelines.

Zowe CLI quick start

Get started with Zowe™ CLI quickly and easily.

Note: This section assumes some prerequisite knowledge of command-line tools and writing scripts. If you prefer more detailed instructions, see [Installing Zowe CLI](#).

- [Installing](#)
- [Issuing your first commands](#)
- [Using profiles](#)
- [Writing scripts](#)
- [Next Steps](#)

Installing

Software Requirements

Before you install Zowe CLI, download and install Node.js and npm. Use an LTS version of Node.js that is compatible with your version of npm. For a list of compatible versions, see [Node.js Previous Releases](#).

(Linux only): On graphical Linux, install `gnome-keyring` and `libsecret` on your computer before you install the Secure Credential Store. On headless Linux, follow the procedure documented in the [SCS plug-in Readme](#).

Installing Zowe CLI core from public npm

Issue the following commands in sequence to install the core CLI.

The "core" includes Zowe CLI and Secure Credential Store, which enhances security by encrypting your username and password.

Installing CLI plug-ins

The command installs most open-source plug-ins, but the IBM Db2 plug-in requires [additional configuration to install](#).

For more information, see [Installing plug-ins](#).

Issuing your first commands

Issue `zowe --help` to display full command help. Append `--help` (alias `-h`) to any command to see available command actions and options.

To interact with the mainframe, type `zowe` followed by a command group, action, and object. Use options to specify your connection details such as password and system name.

Listing all data sets under a high-level qualifier (HLQ)

Downloading a partitioned data-set (PDS) member to local file

See [Command Groups](#) for a list of available functionality.

Using profiles

Zowe profiles let you store configuration details such as username, password, host, and port for a mainframe system. Switch between profiles to quickly target different subsystems and avoid typing connection details on every command.

Profile types

Most command groups require a `zosmf-profile`, but some plug-ins add their own profile types. For example, the CICS plug-in has a `cics-profile`. The profile type that a command requires is defined in the `PROFILE OPTIONS` section of the help response.

Tip: The first `zosmf` profile that you create becomes your default profile. If you don't specify any options on a command, the default profile is used. Issue `zowe profiles -h` to learn about listing profiles and setting defaults.

Creating a zosmf profile

Note: The port defaults to 443 if you omit the `--port` option. Specify a different port if your host system does not use port 443.

Using a zosmf profile

For detailed information about issuing commands, using profiles, and more, see [Using CLI](#).

Writing scripts

You can write Zowe CLI scripts to streamline your daily development processes or conduct mainframe actions from an off-platform automation tool such as Jenkins or TravisCI.

Example:

You want to delete a list of temporary datasets. Use Zowe CLI to download the list, loop through the list, and delete each data set using the `zowe zos-files delete` command.

For more information, see [Writing scripts](#).

Next Steps

You successfully installed Zowe CLI, issued your first commands, and wrote a simple script! Next, you might want to:

- Issue the `zowe --help` command to explore the product functionality, or review the online [web help](#).
- Learn about [using environment variables](#) to store configuration options.
- Learn about [integrating with API Mediation Layer](#).
- Write scripts and integrate them with automation server, such as Jenkins.
- See what [plug-ins are available](#) for the CLI.
- Learn about [developing for the CLI](#) (contributing to core and developing plug-ins).

Information roadmap for Zowe API Mediation Layer

This roadmap outlines information resources that are applicable to the various user roles who are interested in Zowe API Mediation Layer. These resources provide information about various subject areas, such as learning basic skills, installation, developing, and troubleshooting for Zowe API Mediation Layer.

The following definition of skill levels about Zowe assist you with gathering the most relevant resources for you.

- Beginner: You're starting out and want to learn the fundamentals.
- Intermediate: You have some experience but want to learn more in-depth skills.
- Advanced: You have lots of experience and are looking to learn about specialized topics.

Fundamentals

Zowe skill level: Beginner

- [Zowe API Mediation Layer overview](#)

New to API Mediation Layer? This overview topic introduces the key features, main components, benefits, and architecture of API Mediation Layer.

- [Architecture](#)

Review the Zowe architecture to understand how API Mediation Layer works in the Zowe framework.

Installing

Zowe skill level: Beginner

- **System requirements**

Review this topic to ensure that your system meets the requirements for installing the API Mediation Layer. API Mediation Layer is one of the server-side components.

- **Planning**

This article includes details about planning for installation, the Zowe z/OS launch process, and information about the Zowe runtime directory, instance directory, and keystore directory.

- **Installing API Mediation Layer**

This article provides an overview of the essential steps involved in installing API Mediation Layer.

Configuring and updating

Zowe skill level: Intermediate

- **Configuring API Mediation Layer**

- [API Gateway configuration parameters](#)

This article introduces the default configuration of the API Mediation Layer and how to change the parameters to meet your needs.

- [Configuring the Zowe APIs](#)

The article explains how to configure security for the Zowe API Mediation Layer.

- [Advanced Gateway features configuration](#)

This article is for system programmers who want to configure advanced Gateway features of the API Mediation Layer, such as the gateway retry policy, connection limits, gateway timeouts, and so on.

- **Upgrading the z/OS system for Zowe**

Learn how to upgrade the z/OS system for Zowe.

Using Zowe API Mediation Layer

Zowe skill level: Intermediate

- **Using API Catalog**

Learn about how to use the API Catalog to view what services are running in the API Mediation Layer. Through the API Catalog, you can also view the associated API documentation corresponding to a service, descriptive information about the service, and the current state of the service.

- **Blog: Introducing “Try it out” functionality in the Zowe API Mediation Layer**

This blog talks about one key functionality of the Zowe API Mediation Layer to validate that services are returning the expected responses.

- **Docs: Zowe API reference guide**

Find and learn about the Zowe APIs that you can use.

Onboarding APIs

Zowe skill level: Advanced

- **Extend Zowe API Mediation Layer**

Learn how you can extend the Zowe API Mediation Layer. Extenders can build and onboard additional API services to the API ML microservices ecosystem. REST APIs can register to the API Mediation Layer, which makes them available in the API Catalog and for routing through the API Gateway.

- **Onboarding overview**

This article provides details about onboarding a REST API service to the Zowe API Mediation Layer.

- **Zowe API ML repository**

If you want to start working with the code immediately, check out this code repository.

Security

Zowe skill level: Advanced

- **[API Mediation Layer Security](#)**

This article describes how API ML uses Transport Layer Security (TLS). Use this guide to familiarize yourself with the API ML security concepts.

- **[Zowe API Mediation Layer Single-Sign-On Overview](#)**

This article provides an overview of the API ML single-sign-on feature, the principle participants in the SSO process, and links to detailed Zowe SSO documentation.

- **[Blog: The ZAAS Client: a library for the API Mediation Layer](#)**

This blog introduces you to Zowe Authentication and Authorization Service (ZAAS) Client — a library that contains methods for retrieval of JWT tokens, PassTickets, and even verifying JTW token information.

- **[Blog: Single-Sign-On to z/OS REST APIs with Zowe](#)**

This blog takes a deeper dive in to the SSO feature of API ML.

- **[Blog: Zowe client certificate authentication](#)**

Contributing to Zowe API Mediation Layer

Zowe skill level: Advanced

- **[Contributing guidelines](#)**

This document is a summary of conventions and best practices for development within Zowe API Mediation Layer.

- **[Conformance Program](#)**

This topic introduces the Zowe Conformance Program. Conformance provides Independent Software Vendors (ISVs), System Integrators (SIs), and end users greater confidence that their

software will behave as expected. As vendors, you are invited to submit conformance testing results for review and approval by the Open Mainframe Project. If your company provides software based on Zowe CLI, you are encouraged to get certified today.

- **[Blog: Zowe Conformance Program Explained](#)**

This blog describes the Conformance Program in more details.

Troubleshooting and support

- **[Troubleshooting API ML](#)**

Learn about the tools and techniques that are available to help you troubleshoot and resolve problems. You can also find a list of common issues about Zowe API ML.

- **[Error Message Codes](#)**

Use the message code references and the corresponding reasons and actions to help troubleshoot issues.

- **[Submit an issue](#)**

If you have an issue that is specific to Zowe API Mediation Layer, you can submit an issue against the `api-layer` repo.

Community resources

- **[Slack channel](#)**

Join the `#zowe-api` Slack channel to ask questions about Zowe API ML, propose new ideas, and interact with the Zowe community.

- **[Zowe API ML squad meetings](#)**

You can join one of the Zowe API ML squad meetings to get involved.

- **[Zowe Blogs on Medium](#)**

Read a series of blogs about Zowe on Medium to explore use cases, best practices, and more.

- **Community Forums**

Look for discussion on Zowe topics on the [Open Mainframe Project Community Forums](#).

Information roadmap for Zowe Application Framework

This roadmap outlines information resources that are applicable to the various user roles who are interested in Zowe Application Framework. These resources provide information about various subject areas, such as learning basic skills, installation, developing, and troubleshooting for Zowe Application Framework.

The following definition of skill levels about Zowe will help you gather most relevant resources for you.

- Beginner: You're starting out and want to learn the fundamentals.
- Intermediate: You have some experience but want to learn more in-depth skills.
- Advanced: You have lots of experience and are looking to learn about specialized topics.

Fundamentals

Zowe skill level: Beginner

- [Zowe Application Framework overview](#)

New to Zowe Application Framework? This overview topic introduces what is Zowe Application Framework and the main components and benefits.

- [Architecture](#)

Review the Zowe architecture to understand how Zowe Application Framework works in the Zowe framework.

- [An Introduction to the Zowe Virtual Desktop](#)

This blog gives you an overview of the Zowe Desktop.

Installing

Zowe skill level: Beginner

- **System requirements**

Review this topic to ensure that your system meets the requirements for installing the Zowe Application Framework. Zowe Application Framework is one of the server-side components.

- **Planning**

This article provides details about planning for the installation, the Zowe z/OS launch process, and so on.

- **Installing Zowe Application Framework**

This article provides an overview of the essential steps involved in installing Zowe Application Framework. If you want to use Docker for the installation, instead follow the link below.

- **Installing Zowe Application Framework using Docker**

Learn how to use Docker to install Zowe.

Configuring and updating

Zowe skill level: Intermediate

- **Configuring Zowe Application Framework**

This information describes how to configure the Zowe Application Framework as a Mediation Layer client, configure connections for the terminal application plug-ins, modify the Zowe Application Server and Zowe System Services (ZSS) configuration, and so on.

- **Upgrading the z/OS system for Zowe**

Learn how to upgrade Zowe API Mediation Layer to a more recent version.

Using Zowe Application Framework

Zowe skill level: Intermediate

- **Using the Zowe Desktop**

Learn about how to navigate the Zowe Desktop and use a list of pre-installed application plug-ins.

- **Using the Editor**

Learn how to use the Editor application plug-in.

- **Tutorial: Getting started tutorial**

This tutorial walks you through the Zowe Desktop with several simple tasks to help you get familiar with it.

Developing Zowe Desktop plug-ins

Zowe skill level: Advanced

- **Developing for Zowe Application Framework**

Learn how you can extend the Zowe Application Framework by adding a plug-in to the Zowe Desktop.

- **Zowe Application Framework repository**

If you want to start working with the code immediately, check out this code repository.

- **Zowe App Server scripts**

This is the default setup of the Zowe App Server, built upon the zLUX framework. Within, you will find a collection of build, deploy, and run scripts as well as configuration files that will help you to configure a simple App Server and add a few Apps.

Samples

- [Sample iframe App](#)
- [Sample Angular App](#)
- [Sample React App](#)

Contributing to Zowe Application Framework

Zowe skill level: Advanced

- **Conformance Program**

This topic introduces the Zowe Conformance Program. Conformance provides Independent Software Vendors (ISVs), System Integrators (SIs), and end users greater confidence that their software will behave as expected. As vendors, you are invited to submit conformance testing results for review and approval by the Open Mainframe Project. If your company provides software based on Zowe CLI, you are encouraged to get certified today.

- **Blog: Zowe Conformance Program Explained**

This blog describes the Conformance Program in more details.

Troubleshooting and support

- **Troubleshooting Zowe Application Framework**

Learn about the tools and techniques that are available to help you troubleshoot and resolve problems. You can also find a list of common issues about Zowe Application Framework.

- **Submit an issue**

If you have an issue that is specific to Zowe Application Framework, you can submit an issue against the `zlux` repo.

Community resources

- **Slack channel**

Join the Slack channel to ask questions, propose new ideas, and interact with the Zowe community.

- **Zowe WebUI squad meetings**

You can join one of the Zowe WebUI squad meetings to get involved.

- **Zowe Blogs on Medium**

Read a series of blogs about Zowe on Medium to explore use cases, best practices, and more.

- **Community Forums**

Look for discussion on Zowe topics on the [Open Mainframe Project Community Forums](#).

Information roadmap for Zowe CLI

This roadmap outlines the information resources that are available for all user roles who are interested in Zowe CLI. These resources provide information about various subject areas, such as learning basic skills, installation, developing, and troubleshooting for Zowe CLI.

The following definition of skill levels about Zowe will help you gather most relevant resources for you.

- Beginner: You're starting out and want to learn the fundamentals.
- Intermediate: You have some experience but want to learn more in-depth skills.
- Advanced: You have lots of experience and are looking to learn about specialized topics.

Fundamentals

Zowe skill level: Beginner

- [Zowe CLI overview](#)

New to Zowe CLI? This overview topic introduces what is Zowe CLI.

- [Architecture](#)

Review the Zowe architecture to understand how Zowe CLI works in the Zowe framework.

- [Zowe CLI FAQs](#)

If you have a question, it's a good idea to try the FAQ, which answers the most commonly asked questions about Zowe CLI.

Quick start

Zowe skill level: Beginner

- [Zowe CLI quick start](#)

Get started with Zowe CLI quickly and easily.

- [Blog: Getting Started with Zowe CLI](#)

This blog enables you to get started with Zowe CLI quickly.

Installing

Zowe skill level: Beginner

- [System requirements](#)

Review this topic to ensure that your system meets the requirements for installing Zowe CLI.

- [Installing Zowe CLI](#)

Follow the steps to install a new release of Zowe CLI.

Configuring and updating

Zowe skill level: Intermediate

- [Configuring Zowe CLI](#)

This topic explains how to configure Zowe CLI, such as changing log levels and setting the home directory location.

- [Updating Zowe CLI](#)

Learn how to update Zowe CLI to a more recent version using different methods.

Using Zowe CLI and plug-ins

Zowe skill level: Intermediate

- [Using Zowe CLI](#)

Learn about how to use Zowe CLI, including connecting to the mainframe, managing profiles, integrating with API Mediation Layer, and more.

- **Zowe CLI extensions and plug-ins**

This information guides you to explore the extensions and plug-ins available to Zowe CLI and install plug-ins to extend the capabilities of Zowe CLI. Plug-ins add functionality to the product in the form of new command groups, actions, objects, and options.

- **Docs: Zowe CLI command reference guide**

View detailed documentation on commands, actions, and options in Zowe CLI. The reference document is based on the @zowe-v1-lts version of the CLI. You can read an interactive online version, download a PDF document, or download a ZIP file containing the HTML for the online version:

- [Browse online](#)
- [Download CLI reference in PDF format](#)
- [Download CLI reference in ZIP format](#)

- **Best practices**

Are you looking for recommendations and tips on how to best use Zowe CLI to meet your needs? These resources provide best practices recommendations.

- [Blog: Zowe CLI Tips & Tricks](#)
- **Zowe CLI and TSO commands**

This blog shows how to configure and use the TSO command feature of Zowe.

Developing a Zowe CLI plug-in

Zowe skill level: Advanced

Zowe CLI extenders can build plug-ins that provide new commands.

- **Developing for the CLI**

Learn about developing for Zowe CLI.

- [Zowe CLI core repository](#)

If you want to start working with the code immediately, check out this code repository.

- [Tutorials](#)

Follow these tutorials to get started working with a sample plug-in.

Contributing to Zowe CLI

Zowe skill level: Advanced

- [Contributing guidelines](#)

This document is a summary of conventions and best practices for development within Zowe CLI or development of Zowe CLI plug-ins. The guidelines contain critical information about working with the code, running/writing/maintaining automated tests, developing consistent syntax in your plug-in, and ensuring that your plug-in integrates with Zowe CLI properly.

- [Conformance Program](#)

This topic introduces the Zowe Conformance Program. Conformance provides Independent Software Vendors (ISVs), System Integrators (SIs), and end users greater confidence that their software will behave as expected. As vendors, you are invited to submit conformance testing results for review and approval by the Open Mainframe Project. If your company provides software based on Zowe CLI, you are encouraged to get certified today.

- [Blog: Zowe Conformance Program Explained](#)

This blog describes the Conformance Program in more details.

Troubleshooting and support

- [Troubleshooting Zowe CLI](#)

Learn about the tools and techniques that are available to help you troubleshoot and resolve problems. You can also find a list of common issues about Zowe CLI.

- [Submit an issue](#)

If you have an issue that is specific to Zowe CLI, you can submit an issue against the zowe-cli repo.

Community resources

- **Slack channel**

Join the #zowe-cli Slack channel to ask questions about Zowe CLI, propose new ideas, and interact with the Zowe community.

- **Zowe CLI squad meetings**

You can join one of the Zowe CLI squad meetings to get involved.

- **Zowe CLI Blogs on Medium**

Read a series of blogs about Zowe CLI on Medium to explore use cases, best practices, and more.

- **Community Forums**

Look for discussion on Zowe topics on the [Open Mainframe Project Community Forums](#).

Information Roadmap for Zowe Explorer

This roadmap outlines the information resources that are applicable to the users who are interested in Zowe Explorer. These resources provide information about various subject areas, such as learning basic skills, installation, developing, and troubleshooting for Zowe Zowe Explorer.

The following definition of skill levels about Zowe Explorer helps you find the most relevant resources:

- Beginner: You're starting out and want to learn the fundamentals.
- Intermediate: You have some experience but want to learn more in-depth skills.
- Advanced: You have lots of experience and are looking to learn about specialized topics.

Fundamentals

Zowe skill level: Beginner

- [Zowe Explorer overview](#)

New to Zowe Explorer? This overview topic introduces the key features, main components, and benefits of Zowe Explorer.

- [Zowe Explorer FAQs](#)

If you have a question, review the FAQ, which answers the most commonly asked questions about Zowe Explorer.

- [Blog: Visual Studio Code for Mainframe Via the Zowe Explorer Extension](#)

This Medium article outlines the basics of Zowe Explorer, including Getting Started videos.

Installing and configuring

Zowe skill level: Beginner

- [Installing Zowe Explorer](#)

This page includes the system requirements and steps for installing the Zowe Explorer.

- [Video: Getting started with Zowe Explorer \(Part 1\)](#)

- [Video: Getting started with Zowe Explorer \(Part 2\)](#)

These videos help you to get started with Zowe Explorer and show the basic data set use cases.

- [Zowe Explorer Profiles](#)

This page describes how to create and work with Zowe Explorer profiles. Having a profile enables you to use all functions of the extension, activate the Secure Credential Store plug-in to securely store credentials, and more.

Using Zowe Explorer

Zowe skill level: Intermediate

- [Using Zowe Explorer](#)

This page includes usage tips and sample use cases for data sets, USS files, JOBs, and TSO commands. Familiarize yourself with the extension and make the best use of available options and features.

Extending Zowe Explorer

Zowe skill level: Advanced

- [Extend Zowe Explorer](#)

This page describes how to create an extension for Zowe Explorer to introduce more functionalities.

- [Zowe Explorer FTP Extension](#)

This document in Zowe Explorer repository describes how to install and use the FTP extension, which adds the FTP protocol to the Zowe Explorer VS Code extension, allowing you to use z/OS FTP Plug-in for Zowe CLI profiles to connect and interact with z/OS USS.

- **Zowe Explorer repository**

The GitHub repository contains the source code of Zowe Explorer and other Zowe Explorer-related extensions. Check out the landing page README in the repository to find out how to contribute to the extension.

- **Developing for Eclipse Theia**

This article contains information on how to develop for Eclipse Theia.

Contributing to Zowe Explorer

Zowe skill level: Advanced

- **Contributing guidelines**

This document is intended to be a living summary of conventions & best practices for development of the Visual Studio Code Extension for Zowe.

- **Conformance Program**

This topic introduces the Zowe Conformance Program. Conformance provides Independent Software Vendors (ISVs), System Integrators (SIs), and end users greater confidence that their software will behave as expected. As vendors, you are invited to submit conformance testing results for review and approval by the Open Mainframe Project. If your company provides software based on Zowe CLI, you are encouraged to get certified today.

- **Blog: Zowe Conformance Program Explained**

This Medium article provide more details about the Conformance Program, including useful references.

Troubleshooting and support

- **Troubleshooting Zowe Explorer**

Learn about the tools and techniques that are available to help you troubleshoot and resolve problems. You can also find the list of Zowe Explorer issues.

- **Submit an issue**

If you have an issue that is specific to Zowe Explorer, you can submit an issue against the [vscode-extension-for-zowe](#) repository.

Community resources

- **Slack channel**

Join the [# zowe-explorer](#) Slack channel to ask questions, propose new ideas, and interact with the Zowe community.

- **Zowe Explorer squad meetings**

You can join one of the Zowe Explorer squad meetings to get involved.

- **Zowe Blogs on Medium**

Read a series of blog articles about Zowe on Medium to explore use cases, best practices, and more.

- **Community Forums**

Look for discussion on Zowe topics on the [Open Mainframe Project Community Forums](#).

Information roadmap for Zowe Client SDKs

This roadmap outlines information resources that are applicable to the various user roles who are interested in Zowe Client Software Development Kits (SDKs) which is a Zowe component still under development. These resources provide information about various subject areas, such as learning basic skills, installation, developing, and troubleshooting for Zowe Client SDKs.

The following definition of skill levels about Zowe will help you gather most relevant resources for you.

- Beginner: You're starting out and want to learn the fundamentals.
- Intermediate: You have some experience but want to learn more in-depth skills.
- Advanced: You have lots of experience and are looking to learn about specialized topics.

Fundamentals

Zowe skill level: Beginner

- [Zowe Client SDK overview](#)

New to Zowe Client SDKs? This overview topic briefly introduces what it is.

- [Blog: Zowe SDKs - Build z/OS Connected Applications Faster](#)

This blog introduces the Zowe SDKs and their benefits.

Installing

Zowe skill level: Beginner

- [System requirements](#)

Review this topic to ensure that your system meets the requirements for installing Zowe Client SDKs.

- [Installing Zowe SDK](#)

Follow the steps to install Zowe SDKs. You can pull the packages from an online registry, or download the packages from Zowe.org to install locally.

Using Zowe Client SDKs

Zowe skill level: Intermediate

Zowe Node.js SDK

- [Using Zowe Node.js SDKs](#)

This information provides links to different package Readmes that describes how to use the Zowe Node SDK.

- [Docs: Node.js SDK reference guide](#)

Refer to the following Zowe Client SDK reference guides for information about the API endpoints:

- [Browse Node SDK reference guide online](#)
- [Download SDK reference guide in ZIP format](#)

- [Zowe SDK Sample Scripts](#)

This repository contains some sample scripts that utilize various components of the Zowe SDKs organized by use cases.

Zowe Python SDK

- [Using Zowe Python SDKs](#)

This information provides links to different package Readmes that describes how to use the Zowe Python SDK.

- [Docs: Python SDK reference guide](#)

Refer to the following Zowe Client SDK reference guides for information about the API endpoints:

- [Browse Python SDK reference guide online](#)

- [Download SDK reference guide in PDF format](#)

Contributing to Zowe Client SDKs

Zowe skill level: Advanced

- **[Contributing guidelines](#)**

This document is a summary of guidelines for development within Zowe SDKs. You can contribute to add features, enhancements, and bug fixes to the source code.

Troubleshooting and support

- **[Submit an issue](#)**

If you have an issue that is specific to Zowe SDKs, you can submit an issue against the [zowe-cli](#) repo.

Community resources

- **[Slack channel](#)**

Join the #zowe-cli Slack channel to ask questions about Zowe CLI and Zowe SDKs, propose new ideas, and interact with the Zowe community.

- **[Zowe CLI squad meetings](#)**

You can join one of the Zowe CLI squad meetings to discuss Zowe SDKs issues and contribute to Zowe SDKs.

- **[Zowe Blogs on Medium](#)**

Read a series of blogs about Zowe on Medium to explore use cases, best practices, and more.

- **[Community Forums](#)**

Look for discussion on Zowe topics on the [Open Mainframe Project Community Forums](#).

Information roadmap for Zowe Mobile

This roadmap outlines the information resources that are available for all user roles who are interested in the Zowe Mobile Application that is still under incubation. These resources provide information about various subject areas, such as learning basic skills, installation, developing, and troubleshooting for Zowe Client SDKs.

The following definition of skill levels about Zowe will help you gather most relevant resources for you.

- Beginner: You're starting out and want to learn the fundamentals.
- Intermediate: You have some experience but want to learn more in-depth skills.
- Advanced: You have lots of experience and are looking to learn about specialized topics.

Fundamentals

Zowe skill level: Beginner

- [**Blog: Zowe On The Go: A Mobile Application For Modern Mainframers**](#)

This blog by Jessielaine Punongbayan, introduces what is the Zowe mobile application, how it's developed, its architecture, use case and so on.

- [**Video: A Mobile Application For Modern Mainframers**](#)

Watch this video that introduces the Zowe Mobile App in about 1 minute.

- [**Video: The Zowe Mobile Application**](#)

This video gives a demo of what the Zowe Mobile App would look like on your phone.

Installing

Zowe skill level: Beginner

- **Zowe Mobile Application repository**

This README file in the Zowe Mobile App source code repository describes the prerequisites and steps for installing this app.

Community resources

- **Slack channel**

Join the Zowe Slack community to ask questions about Zowe Mobile App, propose new ideas, and interact with the Zowe community.

- **Zowe Blogs on Medium**

Read a series of blogs about Zowe on Medium to explore use cases, best practices, and more.

- **Community Forums**

Look for discussion on Zowe topics on the [Open Mainframe Project Community Forums](#).

Zowe learning resources

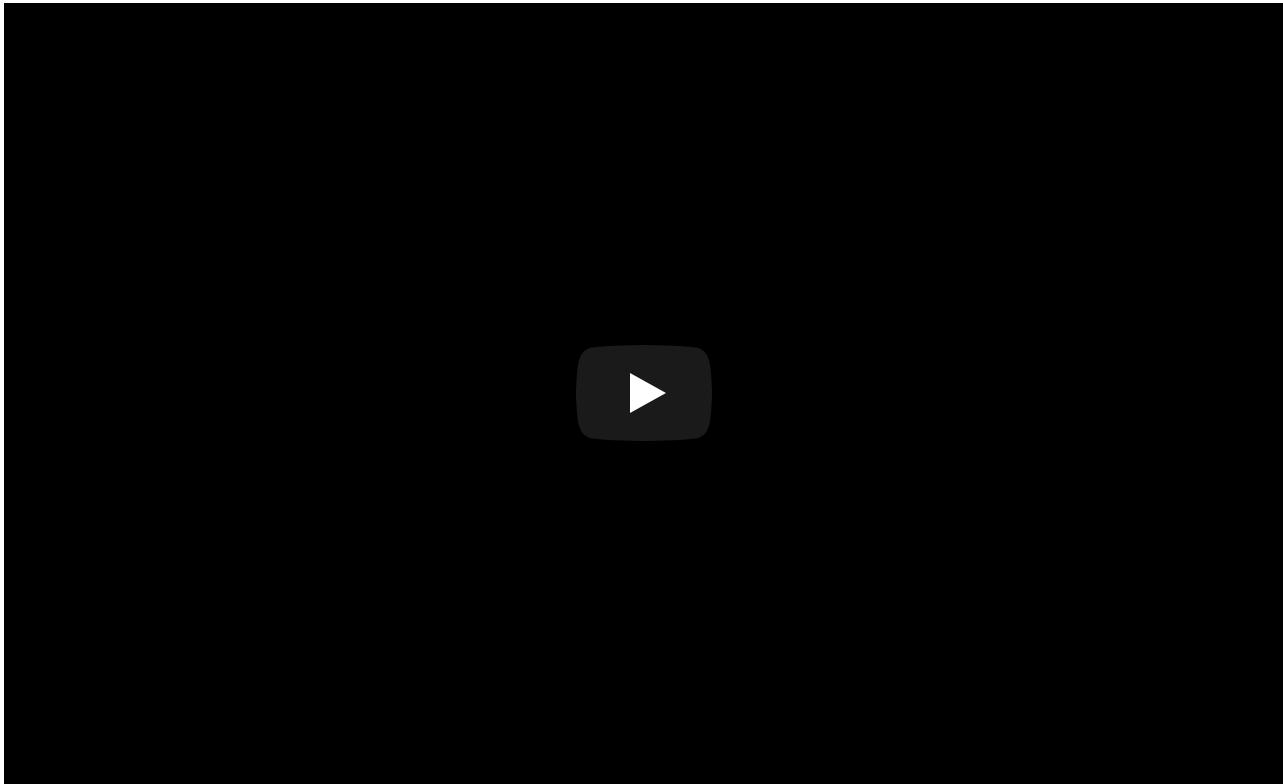
Learn more about Zowe from these blog posts, videos, and other resources.

Blogs

- [Zowe blogs on Medium](#)
- [Zowe blogs on Open Mainframe Project website](#)

Want to contribute a blog? Details for how to contribute to the [Zowe blogs on Medium](#) site are at [Zowe Blog Guidelines](#).

Videos



As well as [Zowe videos](#) owned and managed by the community, there are a number of external youtubers who host Zowe related content.

- [Zowe Demos playlist from Bill Pereira](#)
- [Mainframe Bytes channel from Jessielaine Punongbayan](#)

Webinars

Find out what's happening with Zowe in the Zowe Quarterly Update Webinar Series.

- [Zowe Quarterly Update Webinar: October 2020](#)
- [Zowe Quarterly Update Webinar: January 2021](#)

The [OMP Youtube channel](#) also offers other webinars about Zowe.

- [Treat Yourself to a Guided, Comprehensive Tour of Zowe Desktop Applications](#)
- [Zowe Webinar Feb. 22, 2019](#)
- [Open Mainframe Project Webinar: Zowe Virtual Hackathon](#)

Community

Join us on Slack

- [Slack invite link](#)
- [Introduction to Zowe Slack channels](#)

Learn more about the community

- [Zowe community GitHub repo](#)

Find out information about Zowe sub-projects, GitHub repos, mailing lists, community meeting minutes, contribution guidelines, and so on.

Connect with the community through meetings

- [Zowe meeting calendar](#)

You can join one of the Zowe meetings to get latest Zowe updates and get involved in different squads and initiatives.

Training

Courses

- [Zowe Basics](#)

Interskill Learning offers a free training course that introduces the components that comprise Zowe and the benefits of using Zowe and how its capabilities can be extended.

Trials

- [Zowe trial](#)

The Zowe trial hosted by IBM is a fully configured z/OS environment with Zowe preinstalled and set up along with a set of integrated easy-to-follow tutorials that walk you through the basics of Zowe and gives you hands-on experience of extending Zowe. This no-charge trial is available in two hours for three days.

- [Get started with the Zowe Web UI](#)

This online tutorial hosted by IBM guides you to add new apps to the Zowe Web UI. It provides a public hosted Zowe instance that allows you to perform the steps in a z/OS environment.