

Securely Leverage Open-Source Software with Python AI Toolkit for IBM z/OS

Joe Bostian

Evan Rivera



 Analytics

IBM zSystems

Securely Leverage Open-Source Software with Python AI Toolkit for IBM z/OS

Open-source software (OSS) is widely available and serves as an essential component for enterprises in the artificial intelligence (AI) and machine learning (ML) industry. Specifically, the open-source programming language Python is one of the most versatile and popular programming languages. This situation is especially true in the data science community, where Python provides many libraries and tools that enable essential AI and ML functions, and where it is supported by a large community of developers that actively contribute to its development.

Understanding and managing vulnerabilities within OSS can be complex because of the many components, dependencies, and contributors that are involved. Although the nature of OSS helps balance access to programming and technology, it also results in fast-paced changes to software, which emphasizes the importance of software currency to minimize security concerns. Enterprises understand the critical need to have access to and leverage reputable open-source projects with proper maintenance, updates, transparency, reliable support, and a sense of control to form a secure foundation for implementing AI solutions.

Python AI Toolkit for IBM® z/OS® (see Figure 1) is a powerful set of tools and libraries that is used to establish a secure foundation for AI development and deployment on z/OS so that enterprises can leverage their existing infrastructure for these mission-critical applications. The OSS that is provided within Python AI Toolkit for IBM z/OS is scanned and vetted for security vulnerabilities so that users can make informed decisions when leveraging these Python packages. Packages can be installed and managed by using the Package Installer for Python (pip), which is a common Python package manager, enabling a familiar, flexible, and agile delivery experience while empowering developers to build AI solutions.

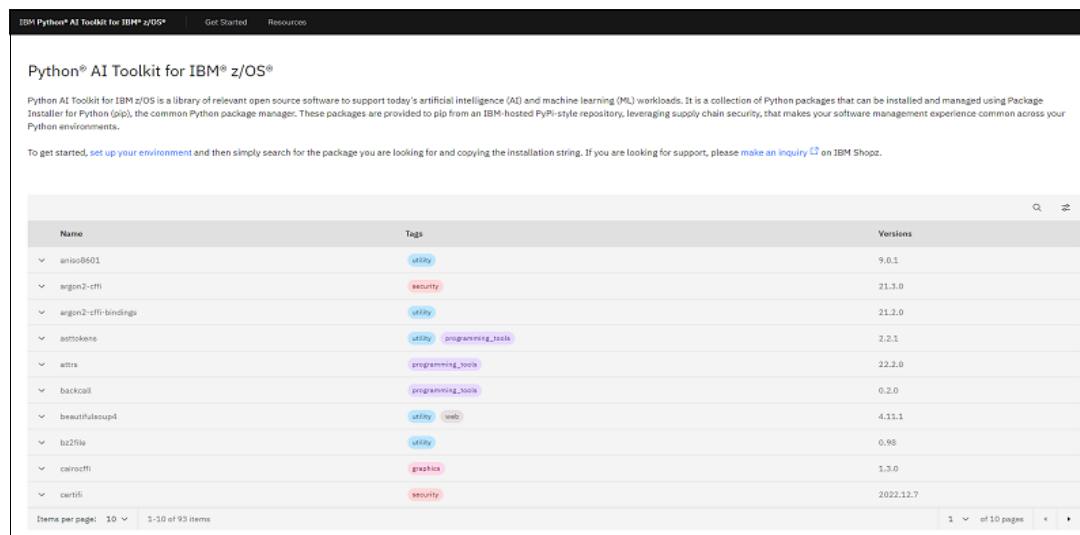


Figure 1 Python AI Toolkit for IBM z/OS repository

Did you know?

The zIIP eligibility list was extended to include Python-based applications, so applications that are built by using the Python AI Toolkit for IBM z/OS benefit by scaling smoothly and remaining cost-effective.

Business value

Python AI Toolkit for IBM z/OS is a part in providing a secure foundation for your AI software stack. With this foundation, you can drive real-time AI insights within mission-critical workloads on IBM zSystems®. Some example applications of AI solutions include fraud detection, anti-money laundering, and image recognition.

Minimizing security exposures and vulnerabilities that might compromise the safety of the operational environment is a priority for enterprises. To mitigate these risks, Python AI Toolkit for IBM z/OS provides the following capabilities:

- ▶ OSS currency that significantly reduces the impact of potential vulnerabilities.
- ▶ All the OSS that is hosted in the library is scanned and vetted for security vulnerabilities by using supply chain security.
- ▶ OSS is delivered through an IBM-owned repository, which means that IBM is the only content contributor and provider. This level of access control helps ensure the reputability of the OSS.

Because OSS is an essential component of the AI ecosystem, and Python is a popular and modern programming language in the AI industry, combining the two is foundational to the development of AI solutions. With the Python AI Toolkit for IBM z/OS, you can develop Python applications within the IBM zSystems environment, which raise the awareness, interest, and feasibility of using modern programming languages to develop and deploy AI applications on IBM zSystems. Your enterprise can appeal to a wider pool of new talent and potentially reduce the skill gap when recruiting and retaining developers.

Solution overview

The Python AI Toolkit for IBM z/OS can be thought of as a runtime library that contains many of the most widely used Python packages for AI and ML workloads, including the following packages:

- ▶ NumPy, SciPy, and Scikit-learn for numerical computation
- ▶ Jupyter, JupyterHub, and Pandas for application development
- ▶ Matplotlib and Cairo for graphics support
- ▶ XGBoost and Apache Tootree (for interacting with Apache Spark from Jupyter) for ML frameworks

Developers can interact with these packages directly by writing their own applications or use them indirectly through products like [IBM Watson Machine Learning for z/OS](#).

Each of the packages in the Python AI Toolkit for IBM z/OS have undergone thorough vetting for known and potential problems by using the [IBM Security and Privacy by Design \(SPbD@IBM\)](#) process that IBM applies to all commercial products and services. Therefore, the OSS that is available within the Python AI Toolkit for IBM z/OS meets the same security requirements as any other IBM product on the z/OS platform.

Security and currency are linked in the open-source domain because most OSS uses point releases for both bug fixes and feature updates, which result in no separation of service streams from new content. This approach requires a user who values security to remain at, or near, the latest release for a OSS package. Python AI Toolkit for IBM z/OS contains over 150 OSS packages, so currency management can become large task.

The Python AI Toolkit for IBM z/OS uses the common Python package manager (pip) to install and maintain the set of OSS packages that make up the product. Administrators can manage Python content from the toolkit by using the same pip CLI that also is on other platforms, which reduces the need for specialized skills. Also, pip can be used manually or incorporated into a workflow for automation, and no unique z/OS skills are required.

Solution architecture

Security considerations are pervasive throughout the design and architecture of the Python AI Toolkit for IBM z/OS product. In addition to properly vetting the OSS packages that make up the Python AI Toolkit for IBM z/OS, it is equally important to secure the channel through which these OSS packages are delivered. The architecture that is outlined in Figure 2 illustrates the general usage of multiple components that enable an administrator to assess and install OSS package sources through any channel from the Python AI Toolkit for IBM z/OS.

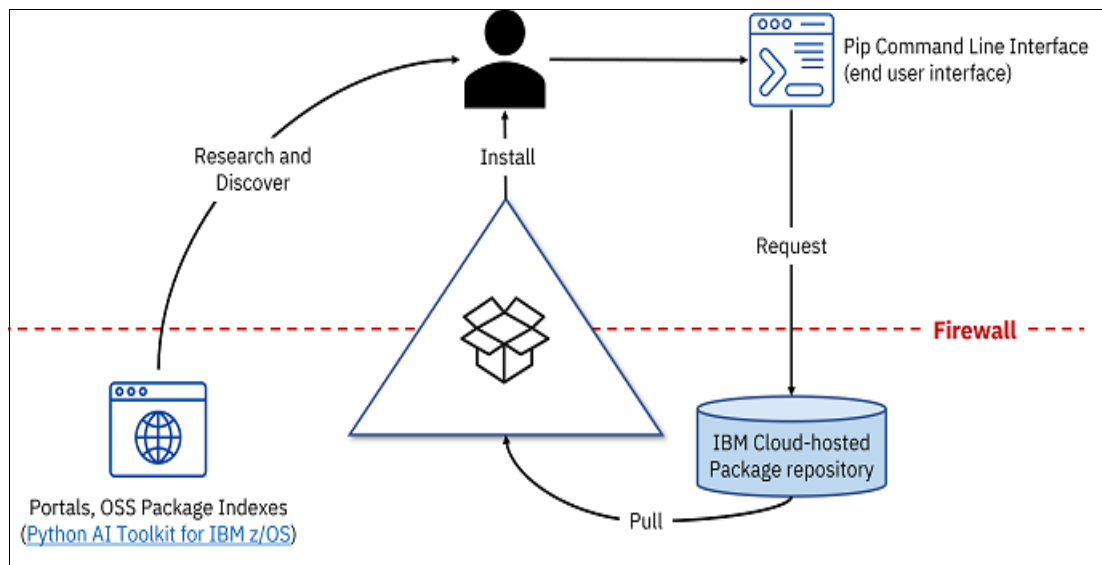


Figure 2 Python AI Toolkit for IBM z/OS channel

Community-based package channels, such as the [Python Package Index \(PyPI\)](#), provide a useful indexes where users can discover new OSS packages and research important details about OSS packages that they might have acquired. PyPI also provides a default repository for pip requests to install a package. These community-based package channels are open to anyone who wants to post a version of a package so that the developer community can share more easily. However, this community support and collaboration also creates a means for threat actors to post content and create a class of problems that is referred to as *Dependency Confusion Attacks*.

The Python AI Toolkit for IBM z/OS provides the same package index and repository architecture as the open environment that is provided by PyPI (see Figure 2 on page 3). However, instead of allowing open contributions to the repository, IBM is the sole provider of OSS packages. Thus, the source code remains open, but the built installable packages are created by IBM, which closes the window of opportunity for threat actors. All these OSS packages are provided to the user through the same pip CLI as on any platform, so using the repository is transparent to those users that manage the Python environment on z/OS.

The collection of vetted OSS packages that is contained within the Python AI Toolkit for IBM z/OS forms a complete closure of the dependency tree, as demonstrated in Figure 2 on page 3, which means that there are no OSS packages that are hosted in the IBM Cloud® package repository that require content to be installed indirectly from another server. This setup prevents unintentional installation of non-vetted code, which reduces the risk of Dependency Confusion Attacks.

Usage scenarios

Python users generally install package content into a local environment and run their applications from it. Users can create configurations that provide a runtime environment that is tailored to a specific application or workflow. This capability is enhanced through a Python facility that is called a [virtual environment](#), which users can use to maintain multiple environments for different applications, which is a common scenario for many enterprises.

Python in an enterprise z/OS context is under stricter governance than in a private workstation, but the same environmental management facilities are useful. Individual z/OS users can install Python OSS packages directly from IBM, and that workflow is more secure than installing OSS packages from a community-based channel, such as PyPI. Most enterprises manage a set of approved OSS packages on an internal repository that mirrors content from a few trusted sources, including IBM. Then, a Python or z/OS system administrator acts as the point of control between the internal managed repository and the outside world. Users can build their local Python environments from the approved internal server. These tasks are done by using the pip CLI, regardless of where the OSS packages come from (PyPI, IBM, or an internal repository). The Python AI Toolkit for IBM z/OS can save enterprises much of the time and resources that are necessary to properly vet OSS for production use.

OSS packages may be installed as a group or individually, depending on the needs of the business. Groups of OSS packages can be managed through a text file that is named `requirements.txt`¹, which contains the list of OSS packages to install (see Example 1). This file also may contain configuration information, like the URL of the server that provides pip with OSS packages during the installation process. The Python AI Toolkit for IBM z/OS publishes these `requirements.txt` files regularly so that administrators can install and upgrade the toolkit as a whole rather than a collection of parts. Administrators also can install individual OSS packages on demand, which is useful if there is a high severity problem that must be patched quickly.

Example 1 Sample requirements.txt with information for installing and maintaining OSS packages

```
#####
#
# Python AI Toolkit for IBM z/OS
#
# This is the complete list of packages that make up the Python AI
# Toolkit for IBM z/OS.
#
# Example use: pip install --no-deps -r <toolkit_requirements.txt>
#
# pip command options:
--index-url
https://downloads.pyaitoolkit.ibm.net:443/repository/python_ai_toolkit_zos/simple
--trusted-host downloads.pyaitoolkit.ibm.net
--require-hashes
--only-binary :all:

#
# This requirements file was generated for a cp311 environment.
#

aniso8601==9.0.1
--hash=sha256:cf9e7fa3cf8f85ed2e99e1aaddff98c27a37c7b3d90c77074aa7415123f66c5a
argon2-cffi==21.3.0
--hash=sha256:ca8776f6acbb0d565b739f1276dea6b6642c3d82d07d5e3456f3a806795eb3ac
argon2-cffi-bindings==21.2.0
--hash=sha256:08a0ec6fa4d22a8b00e82ae7711ef74a50f9717664893b96dc9ebaeb71677e6a
asttokens==2.2.1
--hash=sha256:6b0ac9e93fb0335014d382b8fa9b3afa7df546984258005da0b9e7095b3deb1c
attrs==22.2.0
--hash=sha256:29e95c7f6778868dbd49170f98f8818f78f3dc5e0e37c0b1f474e3561b240836
backcall==0.2.0
--hash=sha256:e0a55adb4d86fa337ea93cf29c79a27b063945a4f8d5ca16122cd0fa9fbc5222
...
```

Often, there are multiple package versions, and sometimes multiple instances of the same package version that are available to users. To mitigate this situation, an additional key pip package management feature that is called `--hash` is available, which a requester can use to install an exact package instance rather than any instance with the right name and version number.

¹ https://github.com/ibm-z-oss-oda/python_ai_toolkit_zos/blob/main/requirements/pyaitoolkit_cp311_2023_0224_req.txt

IBM makes hashes for each package instance available so that users can prevent man-in-the-middle attacks, and other types of spoofing attacks. If there is an unintended misconfiguration or other environmental problem, the installation request fails, which prevents unauthorized code from entering the environment.

The Python community provides a robust set of security measures that the Python AI Toolkit for IBM z/OS uses to help keep your enterprise safe.

Integration

With the Python AI Toolkit for IBM z/OS, you can make various integrations while developing solutions on IBM zSystems, which provides a secure foundation. These integrations permeate the AI stack from the higher-level software offerings to the hardware components and investments for AI and ML.

Opening data analytics

The Python AI Toolkit for IBM z/OS is a key component to derive new insights and benefits from every transaction to open data analytics on IBM zSystems. Other key components open data analytics include IBM Z® Platform for Apache Spark, and IBM Data Virtualization Manager for z/OS. With IBM Z Platform for Apache Spark, you can virtually integrate several different disconnected data sources within a single application. With IBM Data Virtualization Manager for z/OS, you can read/write data to IBM zSystems without transforming or moving it, which makes live data instantly accessible. Leveraging each of these technical components in combination, you can develop world-class data analytics solutions.

AI on the IBM zSystems software stack

The Python AI Toolkit for IBM z/OS is foundational to the overall AI on IBM zSystems software stack because most AI components either are written in the Python language or have hard requirements for key Python OSS packages. The Python AI Toolkit for IBM z/OS provides essential functions that are required in AI solutions higher in the stack. For example, products such as IBM Watson® Machine Learning for z/OS leverage Python OSS packages to implement their functions. These products benefit from obtaining the Python OSS packages that they require from the Python AI Toolkit for IBM z/OS, which are delivered securely with supply chain security.

IBM Open Enterprise SDK for Python 3.10 or later is a requirement when using Python AI Toolkit for IBM z/OS. IBM Open Enterprise SDK for Python provides the installation tool that is needed to pull the Python OSS packages from the IBM repository so that developers can use Python natively on the z/OS platform.

AI on IBM zSystems hardware investments

The Python AI Toolkit for IBM z/OS also benefits from the IBM zSystems hardware investments that are lower in the stack. Acceleration from the IBM Integrated Accelerator for AI provides benefits when running AI workloads that are built on top of the Python AI Toolkit for IBM z/OS. With this workload execution acceleration, enterprises can meet successfully some of the most stringent service-level agreements (SLAs) when integrating AI into business-critical workloads.

Supported platforms

The Python AI Toolkit for IBM z/OS has minimal hardware and software requirements. The software requirements include the following items:

- ▶ The IBM z/OS V2.4 operating system or later
- ▶ IBM Open Enterprise SDK for Python 3.10 or later

The hardware requirements include anything that supports z/OS 2.4 or later, and IBM Open Enterprise SDK for Python 3.10 or later.

Ordering information

The Python AI Toolkit for IBM z/OS is available at [IBM Shopz](#) at no charge for access. There also is IBM Support and Services, which can be added for a fee.

Ordering information is listed in Table 1. For more information, see your IBM representative, IBM Business Partner, or the [Python AI Toolkit for IBM z/OS announcement letter](#).

Table 1 Ordering part numbers and feature codes

Program name	PID number	License option or pricing metric
Python AI Toolkit for IBM z/OS 1.1.0	5698-PAL	Multi-version Measurement No Charge Per Install Qty 1
		Use-Based License One-Time Charge Per Install Qty 1
Python AI Toolkit for IBM z/OS S&S 1.1.0	5698-PLS	Decline Subscribe and Support No Charge Base with Qty 0
		Multi-version Measurement S&S No Charge Per Install Qty 1
		SW Subscription and Support Annual Support Charge Per Install Qty 1
		SW Subscription and Support Monthly Support Charge Per Install Qty 1

Related information

For more information, see the following resources:

- ▶ Getting started with Python AI Toolkit for IBM z/OS video
https://mediacenter.ibm.com/media/t/1_x2q4fh56
- ▶ IBM Offering Information page (announcement letters and sales manuals)
http://www.ibm.com/common/ssi/index.wss?request_locale=en

On this page, enter Python AI Toolkit for IBM z/OS, select the information type, and then click **Search**. On the next page, narrow your search results by geography and language.

- ▶ IBM Z and LinuxONE Journey to Open Data Analytics Content Solution
<https://www.ibm.com/support/z-content-solutions/journey-to-open-data-analytics/>

- ▶ Journey to Open Data Analytics on IBM Z and LinuxONE overview video
https://mediacenter.ibm.com/media/t/1_j1xwphug
- ▶ Overview of Python AI Toolkit for IBM z/OS video
https://mediacenter.ibm.com/media/t/1_ns5n0wtY
- ▶ Python AI Toolkit for IBM z/OS FAQs
https://izswebpage.mybluemix.net/docs/FAQ_zPAIT.pdf
- ▶ Python AI Toolkit for IBM z/OS product page
https://ibm-z-oss-oda.github.io/python_ai_toolkit_zos/

Authors

This IBM Redbooks® Solution Guide was produced by a team of specialists from around the United States.

Joe Bostian is a Senior Technical Staff Member on the AI on IBM zSystems team. He is the Product Owner for the Python AI Toolkit for IBM z/OS. He focuses on infrastructure and solving the key issues regarding open-source deployments on IBM zSystems. This work addresses challenges in packaging, delivery, and open-source security as enterprises become increasingly concerned with threats from both internal and external actors. His goal is to help create a secure deployment architecture that balances with the dynamic characteristics of OSS.

Evan Rivera is a Product Manager in the United States. He has 5 years of experience in the AI and ML field. He holds a master's degree in Computer Science from the Georgia Institute of Technology. He has delivered key AI offerings on IBM zSystems. In addition to product management, he has a technical background as a Machine Learning Engineer and Software Developer. He has experience bringing AI solutions from idea to production.

Thanks to the following people for their contributions to this project:

Makenzie Manna
IBM Redbooks

Shin Kelly Yang, AI on IBM zSystems Product Management
IBM US

Andrew Sica, STSM - AI on IBM zSystems
IBM US

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM Redbooks residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:
ibm.com/redbooks/residencies.html

Stay connected to IBM Redbooks

- ▶ Find us on LinkedIn:
<https://www.linkedin.com/groups/2130806>
- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:
<https://www.redbooks.ibm.com/subscribe>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:
<https://www.redbooks.ibm.com/rss.html>

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

IBM®


IBM Cloud®

IBM Watson®

IBM zSystems®

IBM Z®

Redbooks®

Redbooks (logo) ®

z/OS®

The following terms are trademarks of other companies:

Other company, product, or service names may be trademarks or service marks of others.



REDP-5709-00

ISBN 073846113x

Printed in U.S.A.

Get connected

