

# Release Notes: Junos<sup>®</sup> PyEZ Release 2.0

Release 2.0  
1 September 2016

Junos PyEZ can be used with the following Juniper Networks<sup>®</sup> hardware: ACX Series, EX Series, M Series, MX Series, PTX Series, QFabric systems, QFX Series, SRX Series, and T Series.

These release notes accompany Juniper Networks Junos PyEZ Release 2.0. They describe new and changed features, limitations, and known and resolved problems in the software.

## DISCLAIMER

Use of the Junos PyEZ software implies acceptance of the terms of this disclaimer, in addition to any other licenses and terms required by Juniper Networks.

Juniper Networks is willing to make the Junos PyEZ software available to you only upon the condition that you accept all of the terms contained in this disclaimer. Please read the terms and conditions of this disclaimer carefully.

The Junos PyEZ software is provided "as is." Juniper Networks makes no warranties of any kind whatsoever with respect to this software. All express or implied conditions, representations and warranties, including any warranty of non-infringement or warranty of merchantability or fitness for a particular purpose, are hereby disclaimed and excluded to the extent allowed by applicable law.

In no event will Juniper Networks be liable for any direct or indirect damages, including but not limited to lost revenue, profit or data, or for direct, special, indirect, consequential, incidental or punitive damages however caused and regardless of the theory of liability arising out of the use of or inability to use the software, even if Juniper Networks has been advised of the possibility of such damages.

## Contents

New and Changed Features . . . . .	3
Configuration . . . . .	3
Device Class . . . . .	3
Exception Handling . . . . .	4
Management . . . . .	4
Python Versions . . . . .	5
Changes in Behavior and Syntax . . . . .	5

Known Behavior .....	5
Management .....	5
Known Issues .....	6
Resolved Issues .....	6
Resolved Issues: Release 2.0 .....	6
Resolved Issues: Release 1.3.0 .....	6
Documentation and Release Notes .....	7
Documentation Feedback .....	7
Requesting Technical Support .....	7
Self-Help Online Tools and Resources .....	8
Opening a Case with JTAC .....	8
Revision History .....	8

## New and Changed Features

---

This section describes the new features and enhancements to existing features in Junos PyEZ Release 2.0.

- [Configuration](#)
- [Device Class](#)
- [Exception Handling](#)
- [Management](#)
- [Python Versions](#)

### Configuration

- **Support for JSON configuration data**—Starting with Junos PyEZ Release 2.0, you can load configuration data formatted using JavaScript Object Notation (JSON) on devices running Junos OS Release 16.1 or a later release. To load JSON configuration data, include the **format='json'** argument in the **Config** object **load()** method.

[See [Using Junos PyEZ to Load Configuration Data](#).]

- **Support for using configuration Tables and Views to configure structured resources**—Starting with Junos PyEZ Release 2.0, you can create configuration Tables and Views that include the **set** property to define a structured resource that you can configure on devices running Junos OS. When you add the Table to the Junos PyEZ framework, Junos PyEZ dynamically creates a configuration class for the resource, which you can use to programmatically configure that resource on a device. One benefit of using structured configuration changes to configure a device is that the configuration data does not need to be formatted using one of the accepted formats, such as Junos XML elements.

[See [Overview of Using Junos PyEZ Configuration Tables to Define and Configure Structured Resources](#).]

- **Support for configuration modes**—Starting with Junos PyEZ Release 2.0, when you create a configuration Table object or a **Config** object using a context manager, you can specify the mode in which to make configuration changes. To specify a mode other than the default configuration mode, which updates the shared configuration database, set the **mode** argument equal to **private**, **exclusive**, **dynamic**, or **batch**.

[See [Using Junos PyEZ to Load Configuration Data](#).]

### Device Class

- **New mode parameter**—Starting with Junos PyEZ Release 2.0, you can include the **mode** parameter in the **Device** argument list to specify the connection method to use when connecting to devices running Junos OS. Omit the **mode** argument to cause Junos PyEZ to connect using NETCONF over SSH. Set the **mode** argument to **'serial'** or **'telnet'** to connect using a serial console connection or a telnet connection, respectively. Support for serial console connections and telnet connections through a console server

enable you to use Junos PyEZ to configure new devices that are not yet configured for remote access.

[See [Connecting to Devices Running Junos OS Using Junos PyEZ.](#)]

- **New port parameter**—Starting with Junos PyEZ Release 2.0, you can include the **port** parameter in the **Device** argument list to specify the port to use when connecting to devices running Junos OS. The default value for the **port** parameter depends on the connection type. NETCONF over SSH connections use a default port value of 830. If the **Device** argument list includes **mode='serial'** or **mode='telnet'**, the value for **port** defaults to **/dev/ttyUSB0** and 23, respectively.

[See [Connecting to Devices Running Junos OS Using Junos PyEZ.](#)]

- **Support for feature velocity releases when gathering facts**—Starting with Junos PyEZ Release 2.0, when Junos PyEZ gathers device facts, the **version\_info** field matches on feature velocity release type "F" in addition to the other release types.

## Exception Handling

- **Enhanced RpcError exception**—Starting with Junos PyEZ Release 2.0, the `RpcError` exception has been enhanced so that it extracts and returns all errors and warnings in `<rpc-error>` elements. Prior to this release, Junos PyEZ returned only the first error or warning in the list.

## Management

- **Serial console connection support**—Starting with Junos PyEZ Release 2.0, you can include the **mode='serial'** argument when you create an instance of the **Device** class to connect to a device running Junos OS through a serial console connection. Serial console access enables you to physically connect to and configure a new device that is not yet configured for remote access.

[See [Connecting to Devices Running Junos OS Using Junos PyEZ.](#)]

- **Telnet connection support**—Starting with Junos PyEZ Release 2.0, you can include the **mode='telnet'** argument when you create an instance of the **Device** class to telnet to the device's management interface or to a console server that is directly connected to the device's **CONSOLE** port. Telnet access through a console server enables you to configure a new device that is not yet configured for remote access.

[See [Connecting to Devices Running Junos OS Using Junos PyEZ.](#)]

## Python Versions

- **Support for Python 3.x**—Starting with Junos PyEZ Release 2.0, Junos PyEZ provides support for Python 3.4 and later versions of Python, in addition to the existing support for Python 2.6 and 2.7.

## Changes in Behavior and Syntax

---

This section lists the changes in behavior of Junos PyEZ features and changes in Junos PyEZ syntax from Junos PyEZ Release 2.0.

- **Exceptions for gathering facts result in a warning instead of an error**—Starting with Junos PyEZ Release 2.0, exceptions that occur when gathering facts raise a warning instead of an error, which enables the script to continue running.
- **StartShell utility `run()` method returns a tuple instead of a list**—Starting with Junos PyEZ Release 2.0, the **StartShell** utility `run()` method returns a tuple. The first value of the tuple is a Boolean, which is set to **True** if the command executes successfully with an exit code of 0, and is set to **False** when the exit code is non-zero indicating an error. The second value of the tuple is a string containing the command output. Prior to Junos PyEZ Release 2.0, the `run()` method returned a list where each item in the list corresponded to a single line of output.

## Known Behavior

---

This section contains the known behavior, system maximums, and limitations in software in Junos PyEZ Release 2.0.

## Management

- **Software installation**—The software installation process provided by the `jnpr.junos.utils.sw` module is currently designed to support simple deployment scenarios. The expected use case for this software is deploying new equipment.

The following scenarios are supported:

- Standalone devices with a single Routing Engine
- Standalone devices equipped with dual Routing Engines
- EX Series Virtual Chassis in mixed and non-mixed-mode configurations
- QFX Series Virtual Chassis in mixed and non-mixed-mode configurations
- Mixed EX Series and QFX Series Virtual Chassis
- Deployment configurations that do not have any form of "in-service" features enabled, such as unified ISSU and NSSU

The following scenarios are known *not* to be supported:

- MX Series Virtual Chassis
- SRX Series chassis clusters

- Virtual Chassis Fabric (VCF)
- Deployment configurations that have some form of "in-service" features enabled, such as unified ISSU or NSSU

## Known Issues

---

This section lists the known issues in the software in Junos PyEZ Release 2.0. For the current list of known issues in the Junos PyEZ software, see the open issues listing for the Junos PyEZ project in GitHub at <https://github.com/Juniper/py-junos-eznc/issues>.

- Facts gathering on MX Series routers with dual Routing Engines in a Virtual Chassis configuration only returns information for the master Routing Engine on each member.
- When a Table is instantiated from a file containing XML, Junos PyEZ does not strip out the XML namespaces as it does when a Table is instantiated from a **Device** instance.

## Resolved Issues

---

This section lists the issues fixed in the Junos PyEZ main release and maintenance releases. For the complete and most current list of resolved issues in the Junos PyEZ software, see the closed issues listing for the Junos PyEZ project in GitHub at <https://github.com/Juniper/py-junos-eznc/issues?q=is:issue+is:closed>.

- [Resolved Issues: Release 2.0 on page 6](#)
- [Resolved Issues: Release 1.3.0 on page 6](#)

### Resolved Issues: Release 2.0

- The **Device cli()** method generates an "invalid command" message when the command returns an empty **<rpc-reply>** element.
- Shell sessions created using **StartShell** hang for the root user.
- The **Device probe()** method generates an error when the port is passed in as a string instead of an integer.

### Resolved Issues: Release 1.3.0

- The **jnpr.junos.utils.config.Config** class **commit\_check()** method does not raise an **RpcTimeoutError** when the operation times out.
- The **jnpr.junos.utils.config.Config** class **load()** method throws a **ConfigLoadError** instead of an **RpcTimeoutError** when the operation times out.
- The **jxml.py remove\_namespaces()** function raises an **AttributeError** when encountering comments in the XML.

## Documentation and Release Notes

---

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

## Documentation Feedback

---

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page of the Juniper Networks TechLibrary site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <http://www.juniper.net/techpubs/feedback/>.
- E-mail—Send your comments to [techpubs-comments@juniper.net](mailto:techpubs-comments@juniper.net). Include the document or topic name, URL or page number, and software version (if applicable).

## Requesting Technical Support

---

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

## Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum: <http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

## Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html>.

## Revision History

---

1 September 2016—Revision 1, Junos PyEZ Release 2.0

Copyright © 2017, Juniper Networks, Inc. All rights reserved.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.