

GT 4.2.1 pyGridWare Contribution: System Administrator's Guide

GT 4.2.1 pyGridWare Contribution: System Administrator's Guide

Introduction

This guide contains advanced configuration information for system administrators working with pyGridWare, the python WS-Core. It provides references to information on procedures typically performed by system administrators, including installation, configuring, deploying, and testing the installation.



Important

This information is in addition to the basic Globus Toolkit prerequisite, overview, installation, security configuration instructions in the [Installing GT 4.2.1](#). Read through this guide before continuing!

Table of Contents

1. Dependencies	1
1. Technology Dependencies	1
2. Building and installing	2
3. Configuring	3
1. pyGridWare/config.txt	3
2. pyGridWare/bin/config.txt	3
4. Deploying	4
5. Testing	5
6. Security considerations	6
1. Security Considerations for Python WS Core	6
7. Troubleshooting	7

Chapter 1. Dependencies

1. Technology Dependencies

Required (Use newest version if possible):

- [python 2.3](http://www.python.org)¹
- [pyXML-0.8.4](http://pyxml.sourceforge.net/)²
- [4Suite-1.0a4](http://sourceforge.net/projects/foursuite/)³
- [Twisted-1.3.0](http://www.twistedmatrix.com/products/download)⁴

Optional: WS-Security: XML Digital Signatures, Secure Conversation, etc.

- [pyGlobus](http://dsd.lbl.gov/gtg/projects/pyGlobus/FAQ.html)⁵ Python Wrapper for GT2, needs security and utility modules.
- [Globus 2 Security and utility packages](http://www.globus.org).⁶

¹ <http://www.python.org>

² <http://pyxml.sourceforge.net/>

³ <http://sourceforge.net/projects/foursuite/>

⁴ <http://www.twistedmatrix.com/products/download>

⁵ <http://dsd.lbl.gov/gtg/projects/pyGlobus/FAQ.html>

⁶ <http://www.globus.org>

Chapter 2. Building and installing

Untar the tarball or grab the CVS source, run pyGridWare distutils setup script, optionally regenerate all bindings.

1. Do one of the following:

2. Untar tarball:

```
%tar zxf pyGridWare-1.04b.tar.gz
```

3. Grab the CVS source:

```
%cvs -d :pserver:anonymous@bosshog.lbl.gov:/home/portnoy/u5/repository co login  
%cvs -d :pserver:anonymous@bosshog.lbl.gov:/home/portnoy/u5/repository co pyGridWare
```

4. Change directory

```
%cd pyGridWare
```

5. Run distutils

```
%python setup.py --regenerate install
```

If you want Message Level Security, build the GT2 libraries with a threaded flavor. You can install any GT3 All Source installer bundles. Directions for installing GT 3.2 can be found [here](#)¹.

To build pyGlobus:

1. Set the environment variables *GLOBUS_LOCATION* and *GPT_LOCATION* to the location of the globus installation.
2. Run the pyGlobus distutils setup script:

```
%python setup.py install
```

¹ http://www.globus.org/toolkit/docs/3.2/installation/install_installing.html

Chapter 3. Configuring

There are two configuration files:

1. pyGridWare/config.txt

Read by the distutils setup script when "regenerate" is specified. This is where all service WSDLs are specified. When setup is run, bindings will be created automatically.

[WSDL]

```
CounterService = share/schema/core/samples/counter/counter_service.wsdl
```

2. pyGridWare/bin/config.txt

Configuration parameters for logging, security, container location, and service paths. This file needs to be in the directory from where you are executing the client and/or server.

Chapter 4. Deploying

Run the container script

1. Change to bin directory

```
%cd bin
```

2. Edit the file `server-config.tac` by adding or removing resources:

```
def GetResource(contextFactory=None):  
    root = Resource()  
    root.putChild('wsrf', Resource())  
  
    resource = Resource()  
    root.getStaticEntity('wsrf').putChild('services', resource)  
  
    resource.putChild('CounterService', Counter(post='/wsrf/services/CounterService'))
```

3. Run the start container script:

```
%./start-container.sh
```

Chapter 5. Testing

To test your installation, run:

```
%python unittest
```

Chapter 6. Security considerations

1. Security Considerations for Python WS Core

Individual services can be configured with or without message security, but transport security is a characteristic of the entire container (either using ssl or plain tcp). Authentication and authorization of clients is performed using a callback mechanism.

1.1. Transport Security

Simply edit the file `config.txt` where the executable is being run and turn on ssl.

By default, pyGridWare will look in the user's home directory for the `.globus/usercert.pem` and `.globus/user-key.pem` files.

To use the grid proxy generated by `grid-proxy-init`, just specify the `/tmp/x509***` as the certfile and keyfile.

Example pyGridWare/bin/config.txt

```
[security]
ssl = 1
certfile =
keyfile =
```

Chapter 7. Troubleshooting

1. Make sure all dependencies are met.
2. Building and installing problems:
3. python-2.3 problems We have recently discovered a problem running the `distutils --regenerate` that is caused by a bug in "urllib.basejoin". This is fixed in python2.4. ZSI needs to be patched to fix this problem in python-2.3.

GT 4.2.1 Release Notes: pyGridWare

Table of Contents

1. Component Overview	1
2. Feature Summary	1
3. Changes Summary	1
4. Bug Fixes	1
5. Known Problems	2
6. Technology Dependencies	2
7. Supported Platforms	2
8. Backward Compatibility Summary	2
9. Associated Standards	2
10. For More Information	3

<titleabbrev>Release Notes</titleabbrev>

1. Component Overview

pyGridWare, the python WS Core, provides a basic python toolset for creating WSRF enabled web services, proven to interoperate with the Java WSRF. Performance is a primary concern and motivation. This WSRF project realizes a significant speedup by using C implementations at the performance critical security and DOM layers. WSRF support includes WS-Resource Lifetime, WS-Resource Properties, and WS-Notification. Updated security support for WS-Secure Conversation and Secure Message is also included. Client and service stubs have been generated from the most recent OASIS drafts and WS-Secure Conversation specification, but the bindings can be easily regenerated from WSDL via the setup script at install whenever WSDL needs to be updated.

2. Feature Summary

Features new in release GT 4.2.1:

- [list features]

Other Supported Features

- [list features]

Deprecated Features

- [list features]

3. Changes Summary

No significant changes have been made since the previous stable release.

4. Bug Fixes

No bugs were fixed for Python WS Core.

5. Known Problems

The following problems and limitations are known to exist for Python WS Core at the time of the 4.2.1 release:

5.1. Limitations

- [list limitations]

5.2. Known Bugs

No problems are known to exist for Python WS Core at the time of the 4.2.1 release.

6. Technology Dependencies

Required (Use newest version if possible):

- [python 2.3](#)¹
- [pyXML-0.8.4](#)²
- [4Suite-1.0a4](#)³
- [Twisted-1.3.0](#)⁴

Optional: WS-Security: XML Digital Signatures, Secure Conversation, etc.

- [pyGlobus](#)⁵ Python Wrapper for GT2, needs security and utility modules.
- [Globus 2 Security and utility packages](#).⁶

7. Supported Platforms

Tested Platforms for pyGridWare:

- [list platforms for which component has been tested]

8. Backward Compatibility Summary

This implementation will not be backward compatible with any previous GT3 release.

9. Associated Standards

Associated standards for Python WS Core:

- standard #1

¹ <http://www.python.org>

² <http://pyxml.sourceforge.net/>

³ <http://sourceforge.net/projects/foursuite/>

⁴ <http://www.twistedmatrix.com/products/download>

⁵ <http://dsd.lbl.gov/gtg/projects/pyGlobus/FAQ.html>

⁶ <http://www.globus.org>

- ...
- standard #n

10. For More Information

See [pyGridWare](#) for more information about this component.