Pyro - Python Remote Objects

Oliver Burger

DHBW Stuttgart - Campus Horb

13. Mai 2012

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Was ist Pyro?

DHBW Stuttgart
Duale Hochschule
Baden-Württemberg

Was ist Pyro?

- Eine Python-Bibliothek für den Zugriff auf entfernte Objekte

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Was ist Pyro?

- Eine Python-Bibliothek für den Zugriff auf entfernte Objekte
- Entsprechung zum *Java RMI*

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Was ist Pyro?

- Eine Python-Bibliothek für den Zugriff auf entfernte Objekte
- Entsprechung zum *Java RMI*
- Entwickelt ab 1998

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Was ist Pyro?

- Eine Python-Bibliothek für den Zugriff auf entfernte Objekte
- Entsprechung zum *Java RMI*
- Entwickelt ab 1998
- Reimplementiert 2010

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Funktionsweise

DHBW Stuttgart
Duale Hochschule
Baden-Württemberg

Funktionsweise

- Serialisierung der Objekte mit Hilfe der *pickle*-Bibliothek

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Funktionsweise

- Serialisierung der Objekte mit Hilfe der *pickle*-Bibliothek
- Sicherheitstechnisch problematisch wegen Code-Injections

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Sicherheit

DHBW Stuttgart
Duale Hochschule
Baden-Württemberg

Sicherheit

- Zugang standardmäßig nur für Localhost

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Sicherheit

- Zugang standardmäßig nur für Localhost
- Verschlüsselung der Daten vor der Übertragung

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Sicherheit

- Zugang standardmäßig nur für Localhost
- Verschlüsselung der Daten vor der Übertragung
- Tunnelung der Verbindung über vpn oder ssh

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Sicherheit

- Zugang standardmäßig nur für Localhost
- Verschlüsselung der Daten vor der Übertragung
- Tunnelung der Verbindung über vpn oder ssh
- HMAC-Signatur - Zugriff auf den Server nur für rechtmäßige Clients
  Private-Key-/Public-Key-Verfahren

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Sicherheit

- Zugang standardmäßig nur für Localhost
- Verschlüsselung der Daten vor der Übertragung
- Tunnelung der Verbindung über vpn oder ssh
- HMAC-Signatur - Zugriff auf den Server nur für rechtmäßige Clients Private-Key-/Public-Key-Verfahren
- Ausführen des Servers als eigener Benutzer mit eingeschränkten Rechten

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Server

```
import Pyro4

class GreetingMaker(object):
    def get_fortune(self, name):
        return "Hello, {0}. Here is your fortune message:\n" \
           "Behold the warranty -- the bold print giveth and the " \
           "fine print taketh away.".format(name)

greeting_maker=GreetingMaker()

# create a Pyro daemon
daemon=Pyro4.Daemon()
# register the greeting object as a Pyro object
uri=daemon.register(greeting_maker)

# print the uri so we can use it in the client later
print "Ready. Object uri =", uri
# start the event loop of the server to wait for calls
daemon.requestLoop()
```

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Client

Client

```
import Pyro4

# ask for the uri of the remote object
uri=raw_input("What is the Pyro uri of the greeting object? ") \
  .strip()
name=raw_input("What is your name? ").strip()

# get a Pyro proxy to the greeting object
greeting_maker=Pyro4.Proxy(uri)

# call method normally
print '\n'+greeting_maker.get_fortune(name)+'\n'
```

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Nameserver

DHBW Stuttgart
Duale Hochschule
Baden-Württemberg

Nameserver

- Problem: Objekt-URI muss bekannt sein

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Nameserver

- Problem: Objekt-URI muss bekannt sein
- Ausweg: Pyro-Nameserver

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Nameserver

- Problem: Objekt-URI muss bekannt sein
- Ausweg: Pyro-Nameserver
- Start als Systemdienst über Init-Prozess des OS

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Nameserver

- Problem: Objekt-URI muss bekannt sein
- Ausweg: Pyro-Nameserver
- Start als Systemdienst über Init-Prozess des OS
  nur ein Pyro-Nameserver auf dem Server-PC notwendig

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Änderungen Server

Änderungen Server

```
--- simpleGreeting.py 2012-05-13 12:57:00.130317847 +0200
+++ greeting.py 2012-05-13 13:04:01.545071905 +0200
@@ -10,10 +10,14 @@

 # create a Pyro daemon
 daemon=Pyro4.Daemon()
+# find the name server
+ns=Pyro4.locateNS()
 # register the greeting object as a Pyro object
 uri=daemon.register(greeting_maker)
+# register the object with a name in the name server
+ns.register("example.greeting", uri)

 # print the uri so we can use it in the client later
-print "Ready. Object uri =", uri
+print "Ready."
 # start the event loop of the server to wait for calls
 daemon.requestLoop()
```

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Änderungen Client

Änderungen Client

```
--- simpleClient.py 2012-05-13 12:57:53.717668952 +0200
+++ client.py 2012-05-13 13:05:30.995654686 +0200
@@ -1,13 +1,9 @@
 import Pyro4

-# ask for the uri of the remote object
-uri=raw_input("What is the Pyro uri of the greeting object? ") \
-    .strip()
-
 name=raw_input("What is your name? ").strip()

-# get a Pyro proxy to the greeting object
-greeting_maker=Pyro4.Proxy(uri)
+# use name server object lookup uri shortcut
+greeting_maker=Pyro4.Proxy("PYRONAME:example.greeting")

 # call method normally
 print '\n'+greeting_maker.get_fortune(name)+'\n'
```

DHBW Duale Hochschule Baden-Württemberg Stuttgart

Literatur

- `http://packages.python.org/Pyro4/`, 13.05.2012

DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Ende

Das wär's

DHBW Stuttgart
Duale Hochschule
Baden-Württemberg