Report on IPv6 work at Roma TRE

We are investigating at the moment about the MySQL IPv6 compliance.
Also there are a lot of info about other products that are involved into the middleware that could be now compatible. Also if atm they are not yet tested all together.

So our kind of work will be focused mainly on dbs and at the same time we are contacting people of LSF and Globus to know more news about this.

UDPMON

We report a short description taken from the official UDPMON web site "[…] can be used for measuring the performance of 100 Mbit, Gigabit and 10 Gigabit switches, as well as network connections involving LANs, MANs and WANs. There are also links to results of tests made on Gigabit switches for the ATLAS experiment, and some tests made using the LANs, MANs and WANs of NetNorthWest, Manchester Campus and SuperJANET".

Looking for ipv6 compatibility we have no found references about this. So we have contacted the author, R. Hughes-Jones. He was very interesting in this problem and also if had never tested the software for an effective ipv6 compatibility he knows that some effort is necessary to enable udpmon for ipv6 compatibility.
We have done some tests on the INFN ipv6 site about compliance, but they failed. At the present time , udpmon isn't capable to reach ipv6 addressed hosts.

****

Udpmon is a useful tool designed for testing hardware and network architectures of a specific environment. It can be employed for measurements of the performances of 100 Mbit, Gigabit and 10 Gigabit switches as well network connections of LAN , MAN and WAN. All the scripts included in the distribution can be executed as user , in fact they don't need any root privileges and are specifically implemented for end-to-end performances tests of a network using UDP packets. It can perform different types of measurements:

- Latency as function of packet size using UDP protocol.
- Histograms of latency between a request and a response using UDP protocol.
- Bandwidth measurement , packet loss , corrupted packets , as function of packet size.
- Valuation of the arrival time of received packets.
- Valuation of the one-way-delay between two nodes of the network based on the packet.
- Measurement of user-mode and kernel-mode load for all the CPUs of the transmitting as well receiving hosts on the network.
- Valuation of the interrupts number given by the network card.

CPPUNIT

We can’t define ipv6 compliance for an application like cppunit. It operates at different layer than those which involve internet protocol. In this case , ipv6 compliance is not applicable.

****

CppUnit is a framework for testing C++ code. It avoids the use of inefficient debug methods such as the use of a debugger or control routines written inside the code itself. When configured, cppunit performs all the code tests automatically.

FCGI

Like cppunit , fastCGI operates at different layer than internet protocol. Defining an ipv6 compliance for this module is not applicable.

****

FastCGI is a language independent, scalable, open extension to CGI that provides high performance without the limitations of server specific APIs. The FastCGI interface combines the best aspects of CGI and vendor APIs. Like CGI, FastCGI applications run in separate, isolated processes. FastCGI's advantages include:

- **Performance.** FastCGI processes are persistent-they are reused to handle multiple requests. This solves the CGI performance problem of creating new processes for each request.
- **Simplicity, with easy migration from CGI.** The FastCGI application library (described on page 9) simplifies the migration of existing CGI applications. Applications built with the application library can also run as CGI programs, for backward compatibility with old Web servers.
- **Language independence.** Like CGI, FastCGI applications can be written in any language, not just languages supported by the vendor API.
- **Process isolation.** A buggy FastCGI application cannot crash or corrupt the core server or other applications. A malicious FastCGI application cannot steal any secrets (such as session keys for encryption) from the Web server.
- **Non-proprietary.** FastCGI is supported in all of Open Market's server products, and support is under development for other Web servers, including the freely available Apache and NCSA servers, as well as commercial servers from Microsoft and Netscape.
- **Architecture independence.** The FastCGI interface is not tied to a particular server architecture. Any Web server can implement the FastCGI interface. Also, FastCGI does not impose any architecture on the application: applications can be single or multi-threaded, regardless of the threading architecture of the Web server.

Support for distributed computing. FastCGI provides the ability to run applications remotely, which is useful for distributing load and managing external Web sites.

JERICHO_HTML

Another application that involves different layer from the IP one.Ipv6 compliance is not applicable.

****

Jericho HTML Parser is a simple but powerful java library allowing analysis and manipulation of parts of an HTML document, including some common server-side tags, while reproducing verbatim any unrecognised or invalid HTML. It also provides high-level HTML form manipulation functions. It is an open source library released under the GNU Lesser General Public License (LGPL).

JUG

Jug is an application that works on different layer of the OSI model than Internet Protocol one. Even in this case , ipv6 compliance is not applicable.

****

JUG is a pure java UUID generator, that can be used either as a component in a bigger application, or as a standalone command line tool .UUIDs are 128-bit Universally Unique Identifiers.

JUG generates UUIDs according to the IETF UUID draft specification (and further clarified in UUID URN name space IETF draft) &#8211; all 3 'official' types defined by the draft &#8211; is fast, portable and Open Source (as well as Free Software). You can use JUG in your application according to the license terms of LGPL (Lesser General Public License); or, from version 2.0 on, ASL . More details on this module can be found at Download page .

From version 1.0.0 on, native code (invoked via JNI) for accessing Ethernet MAC address is included with Jug distribution. Note that using this functionality is optional: only time+location - based generation needs MAC address, and even with it, one can just pass the address from a configuration file.

Currently JNI-based Ethernet MAC address support is available on following platforms:

- Linux / x86
- Windows (98, ME, NT, 2K, XP?) / x86
- Solaris / Sparc
- Mac OS X
- FreeBSD / x86

EXIST

The eXist web site http://exist-db.org/index.html doesn’t report any useful information about ipv6 support of the database. We have verified if this software is ipv6 compliant or not , making some test on ipv6 native site at INFN Roma Tre. We have used a testing method very similar to one used for mm-mysql testing procedure. We have chosen the “.war” version of eXist embedded into Tomcat container that we already tested and marked as ipv6 compliant. We have reached successfully the web interface page of eXist from a native ipv6 machine. However we are planning to do more tests on the eXist functionality. We are focusing our attention on more detailed database functionality. We intend to populate an example database and verify if we are able to submit successfully some queries to it.

****

Exist is a native XML open source database that support XQuery and some other correlated standards. It provide a powerful development environment for web implementations based on XQuery standard. They can be realized using XSLT , XHTML ,CSS and Javascript (for AJAX functionalities). XQuery server can be executed from the filesystem or from another database and , in order to avoid an excessive memory load during query processing operations , exist uses an efficiently index structure.

SUNXACML

Ipv6 support in XACMLis not explicitly declared by the authors. In fact it depends by the OASIS standard definitions. Reading documentation on the OASIS web site , we found a confirm concerning the ipv6 XACML compliance. The document , available at http://www.oasis-open.org/committees/download.php/10577/XACML-2.0-OS-ALL.zip (the right one is access_control-xacml-2_0-core-spec-cd-04.pdf) , tell us some useful information about the syntax we can use to implement XACML security policies both on ipv4 and ipv6.

****

Consist of an open source implementation totally rewritten in Java of the OASIS XACML standard. OASIS is an international no-profit consortium that leads development , overlap and use of either public or private e-business standards. XACML consist of an access-control oriented language , based upon XML an standardized by the OASIS consortium. XACML includes a complete language for access-control policies and management of the requests and the responses that occurs in that context. Numerous advantages offers XACML such as application independent standard language for access-control and flexibility.

JAKARTA COMMONS

All the commons belonging to Jakarta project has been marked as ipv6 compliant depending on reasonable considerations. All this modules are based upon java and they work with tomcat container. Since both of them , java and tomcat , are ipv6 compliant , we have reasonably considered that they could have the same compatibility. Up to now they haven’t been tested yet to verify their effective ipv6 support.

COMMONS-CLI

The Jakarta Commons CLI library provides an API for processing command line interfaces. It was formed by the merger of ideas and code from three different libraries - Werken, Avalon and Optz.

There are three stages to command line processing. They are the definition, parsing and interrogation stages. In definition stage each command line must define the set of options that will be used to define the interface to the application. This module uses the Options class as a container for Option instances. The result of the definition stage is an Options instance. The parsing stage is where the text passed into the application via the command line is properly processed. This processing operation occurs according to the rules defined by the parser implementation. The final result of this stage is a CommandLine instance. Finally , the interrogation stage is where the application queries the CommanLine to decide what execution branch to take depending on boolean option and uses the option values to provide the application data. Interrogation stage is implemented in the user code and its final result is that the user code is fully informed of all the text that was supplied on the command line and processed according to the parser and Options rules.

COMMONS-CODEC

Commons Codec provides implementations of common encoders and decoders such as Base64 , Hex and BinaryCodec. In particular , this module provides different implementations of language encoders , digest encoders and network encoders.

COMMONS-COLLECTIONS

The Java Collections Framework was a major addition in JDK 1.2. It added many powerful data structures that accelerate development of most significant Java applications. Since that time it has become the recognised standard for collection handling in Java. Commons-Collections seek to build upon the JDK classes by providing new interfaces, implementations and utilities. There are many features, including buffer interface for collections that have a well defined removal order, like FIFOs , many comparator and iterator implementations , and powerful tools for collections management.

COMMONS-CONFIGURATION

Commons Configuration provides a generic configuration interface which enables an application to read configuration data from a wide variety of sources such as properties files , XML documents, JDBC Datasource , applet and servlet parameters. It provides support even for property list files (extension .plist) , JNDI and system properties. Different configuration sources can be mixed using two dedicated methods , `ConfigurationFactory` and a `CompositeConfiguration`. Additional sources of configuration parameters can be created by using custom configuration objects.

COMMONS-HTTPCLIENT

The java.net package provides basic functionality for accessing resources via HTTP, it doesn't provide the full flexibility or functionality needed by many applications. The Jakarta Commons *HttpClient* component seeks to fill this void by providing an efficient, up-to-date, and feature-rich package implementing the client side of the most recent HTTP standards and recommendations. More details on standards compliance and capabilities of this module can be found at http://jakarta.apache.org/httpcomponents/httpclient-3.x/features.html.

Designed for extension while providing robust support for the base HTTP protocol, the *HttpClient* component may be of interest to anyone building HTTP-aware client applications such as web browsers, web service clients, or systems that leverage or extend the HTTP protocol for distributed communication.

COMMONS-LANG

Lang has a series of string utilities oriented to a wide range of strings manipulation operations and character handling. Functions included in this module , in example , can generate pieces of text and are very useful for random passwords generation.

COMMONS-LOGGING

When writing a library it is very useful to log information. However there are many logging implementations out there, and a library cannot impose the use of a particular one on the overall application that the library is a part of.

The Logging package is an ultra-thin bridge between different logging implementations. A library that uses the commons-logging API can be used with any logging implementation at runtime. Commons-logging comes with support for a number of popular logging implementations, and writing adapters for others is a reasonably simple task.

Applications (rather than libraries) may also choose to use commons-logging. While logging-implementation independence is not as important for applications as it is for libraries, using commons-logging does allow the application to change to a different logging implementation without recompiling code.

Note that commons-logging does not attempt to initialise or terminate the underlying logging implementation that is used at runtime; that is the responsibility of the application. However many popular logging implementations do automatically initialise themselves; in this case an application may be able to avoid containing any code that is specific to the logging implementation used.

MM_MYSQL

MM.MySQL has been tested successfully at INFN Roma Tre ipv6 testbed. We have installed server and client on different computers , then we have created a testing database on the server machine. Finally we have submitted some queries from the client computer and verified the result. All the queries returned the attended result. We can affirm that MM.MySQL module is ipv6 compliant.

****

MM.MySQL is a Type-4 JDBC driver that is under the GNU Library License. It allows Java developers to make connections to MySQL servers from both Java applications and applets.

In addition to the normal JDBC features, MM.MySQL features automatic re-connect of failed connections, Unicode support, and support for varied character encodings.

GLIB2-DEVEL

Glib2-devel includes many headers necessary for all these makefiles that need them to compile. GNOME and other GTK+ based applications use this library. In particular GNOME works over ipv6 as we have tested few time ago installing a graphical environment on a native ipv6 machine. In virtue of all these facts we guess that glib2-devel can provide ipv6 support if it is configured to do that.

****

It is the corresponding Glib development package which includes all the headers necessary to the makefiles of these programs that require it. GLib is a multiplatform library with a wide range of functionalities. It is included into the GTK+ project but is even used by other applications. It is very useful for writing code at low level. It includes a wide range of functionalities such as memory allocation , log systems , string manipulation , type and object system and macros. It provide data structures too such as caching algorithm , string groups , binary balanced tree.