



Managing hypervisors with the Archipel VM orchestration tool

Hyper Management

The innovative Archipel cross-hypervisor management tool uses the Jabber protocol to chat with servers running Xen, KVM, OpenVZ, and VMware. *By Andrej Radonic*



The Open Source community has delivered some of the finest virtualization technologies in the world, including the Xen and KVM hypervisors, various powerful APIs, and interfaces such as Libvirt. But thus far, no open source project has produced a really well-designed cross-hypervisor management tool for virtualized environments. The few candidates have either been restricted to a specific interface, such as the Virt-manager [1] X11 application, or they have been high-flying tools with Cloud aspirations, such as Xen's Enomalism [2], which is over the top for normal hypervisor administration work.

AUTHOR

Andrej Radonic is an IT writer, a member of the board of Intersales AG in Cologne, Germany, and coauthor of the book *Xen 3.2*. His special interests are virtualization, open source enterprise solutions, and e-commerce.

What administrators are increasingly looking for is an intuitive web GUI capable of providing centralized and complete management for thousands of virtual machines from the whole zoo of available systems: Xen, through KVM, through VMware. Enter Archipel [3], with its fresh, new approach: The small team of French developers claims to want to do a better job than Enomalism or Proxmox [4] and delves deeply into the open source toolbox to do so.

The web GUI relies on the Cappuccino JavaScript framework [5] to offer an experience similar to a desktop application. At its core, the Archipel relies on the Libvirt API to enable support for all of the popular hypervisors, such as KVM, Xen, OpenVZ,



and VMware. Just recently, the second beta version of Archipel was released.

Talking to the Hypervisor

Archipel takes a new approach by choosing XMPP as the protocol for talking to the hypervisor. This instant messaging protocol, which was made popular by Jabber, supports internal messaging for control and information purposes in real time. This means you can even use a Jabber client to query status information from virtual machines or to send commands. At its core, Archipel comprises three components (Figure 1):

- The Archipel client provides an attractive web GUI that is highly reminiscent of iTunes and that will run in most browsers, thanks to JavaScript. (Test clients are available [6].)
- An Archipel agent is used on the server to reach hypervisor instances. The agent is written in Python and has a modular structure. Administrators can use shell commands to install and update the agent.
- An XMPP server setup is necessary because the agent uses XMPP with Libvirt underpinnings to talk to the virtual machines.

Archipel recommends the Ejabberd Jabber/XMPP server [7], although other implementations are said to work. A command to start a virtual machine passes through Jabber XMPP on the Archipel client to the XMPP server; it then reaches Libvirt (Libvirt-create), and finally the hypervisor (e.g., KVM-create).

Installation

One of the biggest disadvantages of Archipel is revealed when you come to the installation: the extremely sparse documentation. But at least the description of the setup is pretty coherent [8], although you need to watch out for a couple of bugs. The best approach is to start by installing Ejabberd, which needs to be at least version 2.1.6, preferably using the binary installer.

Next, you need to extend the XMPP server by adding the `ejabberd_xmlrpc` and `mod_admin_extra` modules. To do so, retrieve the source code, which is written in Erlang, then copy the resulting `*.beam` files to the Ejabberd daemon's `ebin` directory (Listings 1 and 2).

After completing this step, you need to modify `/opt/ejabberd-2.1.6/conf/ejabberd.cfg` (depending on your distribution this file might be `/etc/ejabberd/ejabberd.cfg`). To do so, enable modules that you just created and, above all, enter the fully qualified domain name (FQDN). Unfortunately, the configuration approach documented by the Archipel project is misleading in various places; for this reason, you should probably keep to Listing 3.

LISTING 1: ejabberd_xmlrpc

```
wget http://www.ejabberd.im/files/contributions/xmlrpc-1.13-ipr2.tgz
tar -xzf xmlrpc-1.13-ipr2.tgz
cd xmlrpc-1.13/src
make
cd ../../
cp ebin/*.beam /opt/ejabberd-2.1.6/lib/ejabberd-2.1.6/ebin
```

Finally, create an administrator account to complete the Ejabberd setup:

```
ejabberdctl register admin FQDN password
```

The process of setting up the Archipel agent is somewhat quicker and starts with a dependency check, for which you need Python version 2.5 or newer, Libvirt version 0.8.7 or newer, and a hypervisor such as KVM with Qemu version 0.12.5 or newer. Additionally, you should have Qemu-img and the Python setup tools.

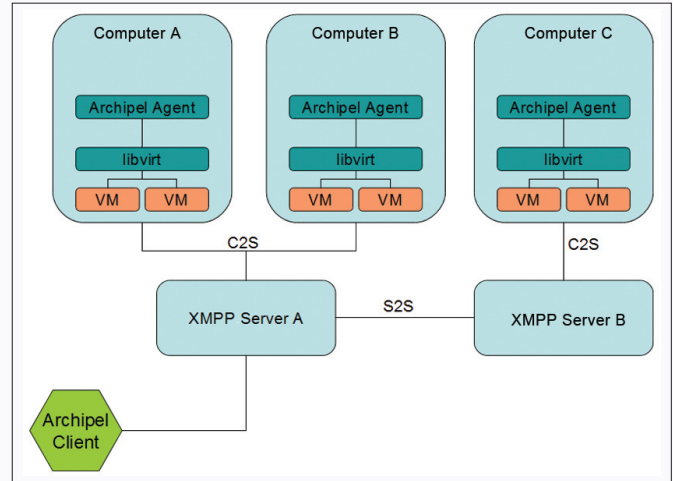


Figure 1: The Archipel architecture with hypervisors, virtual machines, Archipel agents, and redundant XMPP servers.

LISTING 2: mod_admin_extra

```
cd /usr/local/src/ejabberd-modules/mod_admin_extra/trunk/
./build.sh
cp ebin/mod_admin_extra.beam
/opt/ejabberd-2.1.6/lib/ejabberd-2.1.6/ebin
```

LISTING 3: ejabberd.cfg

```
01 {hosts, ["jabber.deutschewolke.datenwerk-it.de"]}.
02 [...].
03 {listen,
04 [
05 {4560, ejabberd_xmlrpc, []},
06 {5280, ejabberd_http, [
07 http_bind,
08 http_poll,
09 web_admin
10 ]}
11 ]}.
12 [...].
13 {modules,
14 [
15 {mod_adhoc, []},
16 {mod_http_bind, []},
17 [...].
18 {mod_admin_extra, []}
19 ]}.
20 [...]
```

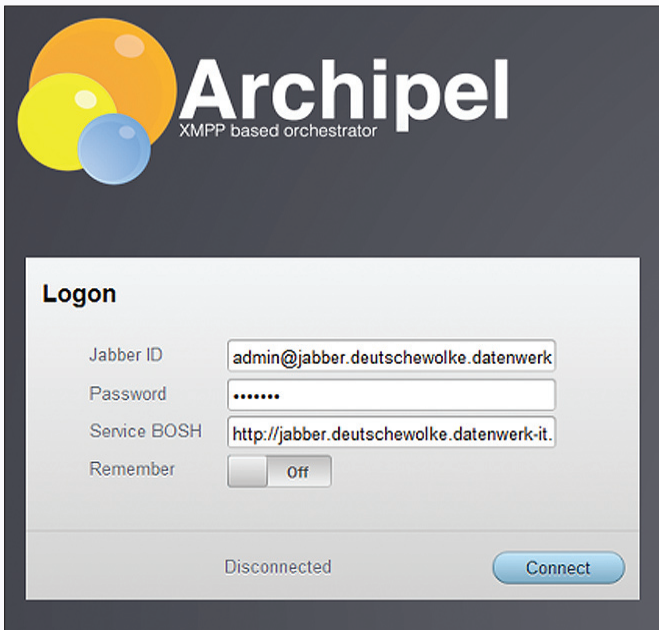


Figure 2: For an Archipel login on the client, you need the complete login name (including the domain) in the Jabber ID field.

The following sequence of commands installs the Archipel agent:

```
easy_install archipel-agent
archipel-initinstall
```

Now you need to add the server FQDN to `/etc/archipel/archipel.conf` and launch Archipel using the `init` script provided:

```
/etc/init.d/archipel start
```

Finally, you need two Ejabberd pubsub nodes to make sure the rights system and tag management will work:

```
archipel-tagnode --jid=admin@FQDN --password=Password --create
SUCCESS: pubsub node /archipel/tags created!

archipel-rolesnode --jid=admin@FQDN --password=Password --create
SUCCESS: pubsub node /archipel/roles created!
```

You won't need to build the clients manually, as described in great detail by the Archipel wiki. Instead, just pick up the latest Archipel client distribution [9], unpack it in a local directory, and open `index.html` in your browser. For the login, it is important to enter the fully qualified Jabber ID; that is, the ID has to include the complete server name (Figure 2). The password you need here is the credential you specified when creating the administrator account. The URL for the *Service BOSH* field follows a pattern of `http://FQDN:5280/http-bind`.

Finding Your Virtual Machines

Archipel comes up with a tidy user interface that shows a great deal of attention to detail. But if you don't populate your environment with hypervisor hosts and virtual machines, even the prettiest of all interfaces will not be much help. In the first step toward populating your environment, you have to register the hypervisor with Archipel. To do so, click on the + symbol in the bottom left to create a new (Jabber) contact on the Archipel XMPP system.

The contact ID, in this case, is `Hypervisor@FQDN`; click *OK* when you are done. The hypervisor and its avatar appear in your contact list on the left. The main view of the host is the Health view, which gives you a full set of critical information in real-time (Figure 3).

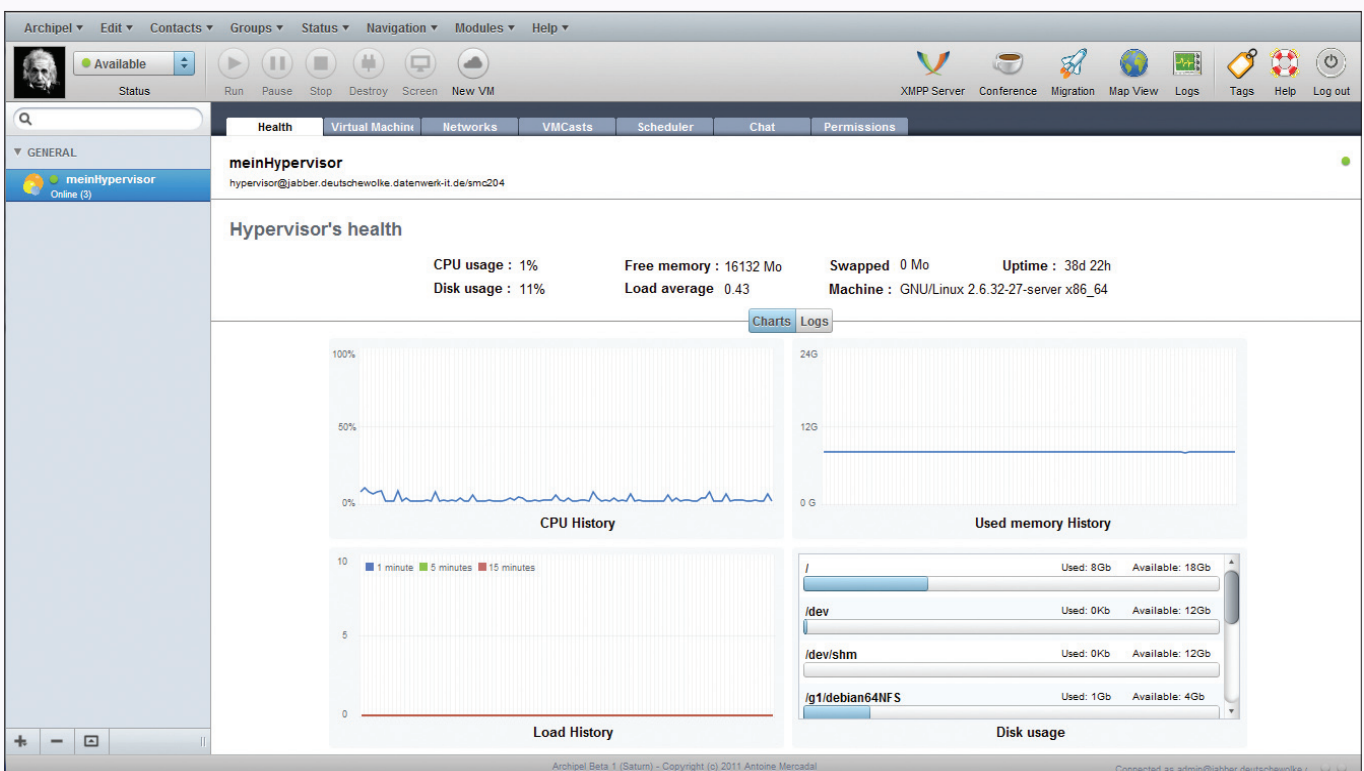


Figure 3: Checking the hypervisor's wellness in Archipel's health summary.

Unfortunately, Archipel still hasn't been introduced to this hypervisor's virtual machines. You have two options: You can add an existing virtual machine using the Archipel management tools; the `archipel-importvirtualmachine` script is helpful here. The script expects two arguments, and you have to find out the correct values up front. One of the arguments is the SQLite 3 database description file for the local hypervisor, which is located in `/var/lib/archipel/hypervisor.sqlite3.2`; the other argument is the UUID of the virtual machine you want to import. The easiest way of discovering the UUID is to run the `virsh` Libvirt tool:

```
virsh --connect qemu:///system
list
dominfo ID_of_VM
UUID
```

The UUID is necessary to start the import process (Listing 4). The virtual machine will now also appear in the contact list on the left, and it is here that the Archipel manager can view the machine status and execute life-cycle commands – such as `start`, `stop`, and `pause` – and even migrate the virtual machine to another hypervisor host or create snapshots.

The second option for increasing your virtual machine population in Archipel is to create a virtual machine directly in the Archipel GUI. This process involves just a couple of steps. To begin, store the required installation ISO image in the `/vm/iso` directory. Then, in the GUI, click on *New VM*, go to the *Definition* tab, and enter the required settings for RAM, disk drives, and the network (Figure 4).

After doing this, you need to create a new (virtual) CD-ROM drive and set the virtual machine to boot from CD. Clicking on *Play* will wake up the virtual machine and trigger the installation. The JavaScript implementation of VNC in the *VNC Console* tab provides a lean and attractive approach to managing the virtual machine's graphical interface.

Talking to a Virtual Machine

One original detail, which shows how consistently Archipel builds on the XMPP protocol, is integrated chat. Of course, you can use Chat to communicate with other administrators in your environment, but surprisingly, it has an option for direct, verbal communication with the hypervisors and virtual machines (Figure 5).

Archipel's natural language vocabulary is fairly restricted, but communicating in this way is fun and surprisingly capable in the production environment. If you ask the virtual machine, *how are you*, it will reply with a compressed status message in the first person singular. You can use any Jabber client to chat in this way. Archipel is always happy to give you information.

LISTING 4: Importing VMs

```
/etc/init.d/archipel stop # Archipel Agent stop
archipel-importvirtualmachine
--file=/var/lib/archipel/hypervisor.sqlite3
--uuid=UUID --xmppserver=FQDN --name=vm1
/etc/init.d/archipel start # Archipel Agent start
```

Online Training with Linux Magazine Academy Preparing for the LPIC exam - the easy way:

Equivalent to CompTIA Linux+
powered by LPI

GET YOUR LINUX KNOW-HOW CERTIFIED WITH LPIC:

Professional training
for the exams
LPI 101 and 102



- hardware settings
- package management
- partitioning and filesystems
- shell environment
- automate system administration
- network configuration
- security administration tasks
- troubleshooting

20%
off for current
subscribers

For more information and to order:
academy.linux-magazine.com/LPIC

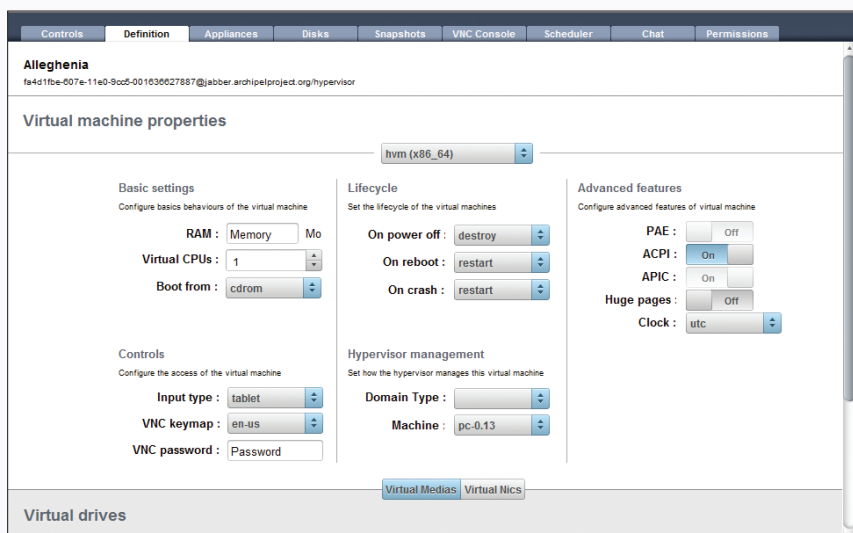


Figure 4: Configuring a virtual machine is self-explanatory.

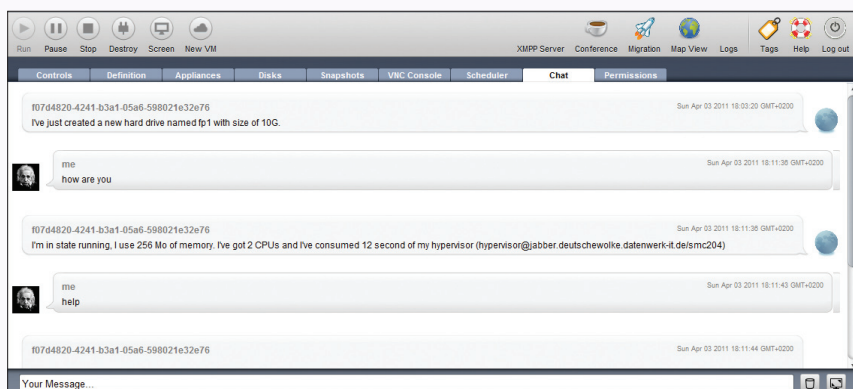


Figure 5: Archipel responds to questions in the first person singular and is very happy to give you details of the hypervisor and virtual machine health states.

Administrators can view logs for all of the systems and their corresponding actions in real time in the GUI; you can also push selected messages to your smartphone. To do so, just load, for example, the App Notifications app on your iPhone and enter the private API key in `archipel.conf`.

Role Concepts, Deployment, Tagging

As a full-fledged management solution for virtual machines, Archipel has an arsenal of powerful deployment tools: The integrated VM Casting Protocol (originally developed by Enomalism) uses RSS feeds to support appliance downloads for the hypervisors. This means you can deploy virtual machine instances and updates automatically on the basis of defined appliances.

Additionally, Archipel administrators can create appliances directly from the virtual machines themselves, store the results as templates, and use them for new virtual machines. Archipel supports the XVM2 template format (which was also introduced by Enomalism and is known by XenServer) for this and will support the popular OVF standard in the future.

Many management tools stop with this level of functionality, but Archipel offers several additional enterprise features. These enhanced features include a sophisticated role concept for defining granular privileges for different users and roles. Administrators of very large environments with large numbers of vir-

tual machines will appreciate the tagging system and the corresponding search feature, as well as the ability to group contacts logically – that is, hypervisors and virtual machines.

The ability to control a whole group of virtual machines by means of mass commands (e.g., starting, stopping, or migrating multiple virtual machines at the same time) is very useful. Archipel also allows you to schedule actions that relate to your virtual machines in advance and automate the process.

Clusters, Geographical Migration, and Billing

To avoid a single point of failure with a single XMPP server, Archipel supports clustering. In fact, it supports setting up multiple sites, and it is capable of visualizing the hypervisor hosts, as well as migrating the virtual machines that reside on them graphically in an integrated Google map. The road-map also includes a module extension for cloud computing capabilities, including billing.

As soon as the promised SDK is released, nothing will prevent modular, customized development of Archipel-based management systems. It is just a pity that the final version probably won't be available until the end of 2011. If you enjoy experimenting and have some time to spare, you can perform a test, but you need to be patient and not be put off

by the very sparse documentation. One hopes, the budding Archipel community will grow quickly and get rid of its teething pains thanks to external contributions.

As of this writing, Archipel is already far more than just a neat GUI for open source hypervisors. Administrators in heterogeneous virtual landscapes will find that it is a very interesting and powerful alternative to other web-based management tools.

If you enjoy IRC chat, you will discover that advice and support is quickly forthcoming from the community – and even from the developers themselves, who gave me some very useful support when I was writing this article. ■■■

INFO

- [1] Virt-manager: <http://virt-manager.et.redhat.com>
- [2] Enomalism: <http://wiki.xensource.com/xenwiki/Enomalism>
- [3] Archipel: <http://archipelproject.org>
- [4] Proxmox VE: <http://www.proxmox.com>
- [5] Cappuccino framework: <http://cappuccino.org>
- [6] Test clients for Archipel: <http://app.archipelproject.org>
- [7] Ejabberd: <http://www.ejabberd.im>
- [8] Archipel wiki: <https://github.com/primalmotion/archipel/wiki>
- [9] Nightly builds: <http://nightlies.archipelproject.org>