Introduction to the Pakfire Build Service

The new IPFire build system

Michael Tremer

IPFire Project

April 28, 2012

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

The Pakfire Build Service is the central build system of the IPFire firewall distribution.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first...Terminology
Conclusion

It automates the whole process from source code to a binary package that can be easily installed on any IPFire system.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

What is a package?

In general, a package is a collection of files and their metadata.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

What is a package?

In general, a package is a collection of files and their metadata.

It is a single file and hence can easily be sent around for further development or distribution. File Extension: pfm.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

What is a package?

In general, a package is a collection of files and their metadata.

It is a single file and hence can easily be sent around for further development or distribution. File Extension: pfm.

There are two different types of packages…

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

What is a package?

In general, a package is a collection of files and their metadata.

It is a single file and hence can easily be sent around for further development or distribution. File Extension: pfm.

There are two different types of packages…

- Source packages

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

What is a package?

In general, a package is a collection of files and their metadata.

It is a single file and hence can easily be sent around for further development or distribution. File Extension: pfm.

There are two different types of packages…

- Source packages
- Binary packages

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Source packages

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Source packages

- A source package contains source code and build instructions to compile the sources.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Source packages

- A source package contains source code and build instructions to compile the sources.
- Source code is considered to be the release tarball of an upstream project and a number of patches.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Source packages

- A source package contains source code and build instructions to compile the sources.
- Source code is considered to be the release tarball of an upstream project and a number of patches.
- The result after putting a source package into a build is one or more binary packages.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Binary packages

Binary packages are ready for installation on target systems.

It's easy:

pakfire install htop-1.0.1-1.ip3.x86_64.pfm

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Binary packages: Example

audit - User space tools for 2.6 kernel auditing

- audit-2.2-2.ip3.src.pfm

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first...Terminology
Conclusion

Binary packages: Example

audit - User space tools for 2.6 kernel auditing

- audit-2.2-2.ip3.src.pfm

The source package produces a whole set of binary packages:

- audit-2.2-2.ip3.x86_64.pfm
- audit-debuginfo-2.2-2.ip3.x86_64.pfm
- audit-devel-2.2-2.ip3.x86_64.pfm
- audit-libs-2.2-2.ip3.ip3.x86_64.pfm

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

Packages have different kinds of *dependencies* which define relationships to each other.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

Packages have different kinds of *dependencies* which define relationships to each other.

A dependency can be defined as:

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

Packages have different kinds of *dependencies* which define relationships to each other.

A dependency can be defined as:

- An other package by name (*openssl, beep or zlib-devel*).

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

Packages have different kinds of *dependencies* which define relationships to each other.

A dependency can be defined as:

- An other package by name (*openssl, beep or zlib-devel*).
- An other package by name and version (*glibc >= 2.15 or udev < 182*).

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

Packages have different kinds of *dependencies* which define relationships to each other.

A dependency can be defined as:

- An other package by name (*openssl, beep or zlib-devel*).
- An other package by name and version (*glibc >= 2.15 or udev < 182*).
- A filename like */usr/bin/bash*.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

Packages have different kinds of *dependencies* which define relationships to each other.

A dependency can be defined as:

- An other package by name (*openssl, beep or zlib-devel*).
- An other package by name and version (*glibc >= 2.15 or udev < 182*).
- A filename like */usr/bin/bash*.
- A string that is provided by any package (e.g. *bootloader*). You should not use this.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

A given package *X* may have the following dependencies:

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

A given package *X* may have the following dependencies:

- *Requires*: The installation of package *X* requires the availability of everything in this list.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

A given package *X* may have the following dependencies:

- *Requires*: The installation of package *X* requires the availability of everything in this list.
- *Provides*: The package in hand comes with the features in this list.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

A given package *X* may have the following dependencies:

- *Requires*: The installation of package *X* requires the availability of everything in this list.
- *Provides*: The package in hand comes with the features in this list.
- *Conflicts*: Package X cannot be installed if any of these dependencies is provided by an other package.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Packages have got dependencies

A given package *X* may have the following dependencies:

- *Requires*: The installation of package *X* requires the availability of everything in this list.
- *Provides*: The package in hand comes with the features in this list.
- *Conflicts*: Package X cannot be installed if any of these dependencies is provided by an other package.
- *Obsoletes*: If package X is installed, there is no need for anything in this list.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For users:

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For users:

- Install only what you need.
    - Keeps the system small and fast.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For users:
- Install only what you need.
  - Keeps the system small and fast.
- Better updates:
  - Update only what you want (e.g. only security updates).
  - Faster distribution of hotfixes.
  - Rollback bad updates.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For users:

- Install only what you need.
    - Keeps the system small and fast.
- Better updates:
    - Update only what you want (e.g. only security updates).
    - Faster distribution of hotfixes.
    - Rollback bad updates.
- Use other package sources.
    - Third party repositories for closed source stuff.
    - Packages we don't want to have in IPFire.
    - Whatever you might think of...

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For developers:

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For developers:

- Each developer gets a small chunk to maintain.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For developers:

- Each developer gets a small chunk to maintain.
- One broken package does not break the whole distribution.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For developers:

- Each developer gets a small chunk to maintain.
- One broken package does not break the whole distribution.
- Faster builds:
    - Just rebuild one package.
    - Never the whole distribution.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For developers:

- Each developer gets a small chunk to maintain.
- One broken package does not break the whole distribution.
- Faster builds:
  - Just rebuild one package.
  - Never the whole distribution.
- Send unstable updates to each other. Test them.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Benefits of the package division

For developers:

- Each developer gets a small chunk to maintain.
- One broken package does not break the whole distribution.
- Faster builds:
    - Just rebuild one package.
    - Never the whole distribution.
- Send unstable updates to each other. Test them.
- Push out updates earlier:
    - No release planning for core updates.
    - Don't bother with issues that have already been fixed.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Conclusion

Packages have a lot of advantages.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Conclusion

Packages have a lot of advantages.

But the management of more than 1250 packages per architecture is a tough job.

Introduction
The Pakfire Build Service
Learning by example
The end

What is the Pakfire Build Service?
First things first…Terminology
Conclusion

Conclusion

Packages have a lot of advantages.

But the management of more than 1250 packages per architecture is a tough job.

We need a management tool for that!

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

The Pakfire Build Service (PBS)

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

The Pakfire Build Service (PBS)

- It has been developed by developers of the IPFire project to achieve a very high level of quality in the build process in conjunction with the community.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

The Pakfire Build Service (PBS)

- It has been developed by developers of the IPFire project to achieve a very high level of quality in the build process in conjunction with the community.
- Written in pure Python.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

The Pakfire Build Service (PBS)

- It has been developed by developers of the IPFire project to achieve a very high level of quality in the build process in conjunction with the community.

- Written in pure Python.

- Comes with a web user interface and an XMLRPC management interface.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

The Pakfire Build Service (PBS)

- It has been developed by developers of the IPFire project to achieve a very high level of quality in the build process in conjunction with the community.

- Written in pure Python.

- Comes with a web user interface and an XMLRPC management interface.

- Heavily based on the Pakfire package management system.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

Excursion: pakfire

The Pakfire package management system manages installations, updates, removes and more of packages on an IPFire system.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

Excursion: pakfire

The Pakfire package management system manages installations, updates, removes and more of packages on an IPFire system.

- Written in Python with performance critical code in C.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

Excursion: pakfire

The Pakfire package management system manages installations, updates, removes and more of packages on an IPFire system.

- Written in Python with performance critical code in C.
- Fast and most robust dependency solving algorithm in the world.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

Excursion: pakfire

The Pakfire package management system manages installations, updates, removes and more of packages on an IPFire system.

- Written in Python with performance critical code in C.
- Fast and most robust dependency solving algorithm in the world.
- Very tiny. Runs on embedded systems.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

Excursion: pakfire

The Pakfire package management system manages installations, updates, removes and more of packages on an IPFire system.

- Written in Python with performance critical code in C.
- Fast and most robust dependency solving algorithm in the world.
- Very tiny. Runs on embedded systems.
- Built-in build system.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

More definitions

Package

Primarily, the term *package* describes a source package. It also can rarely mean the actual package file.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

More definitions

Package

Primarily, the term *package* describes a source package. It also can rarely mean the actual package file.

Build

A build is an ordering unit which contains a source package and multiple *build jobs*. There is also some meta information about the reason of the build (commit message, associated bugs).
A build is what we are managing here. We don't bother about a single package file.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

More definitions

Package

Primarily, the term *package* describes a source package. It also can rarely mean the actual package file.

Build

A build is an ordering unit which contains a source package and multiple *build jobs*. There is also some meta information about the reason of the build (commit message, associated bugs).
A build is what we are managing here. We don't bother about a single package file.

Build job or just: job

A build contains several jobs. One for each architecture the build is built for.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

More definitions

Package

Primarily, the term *package* describes a source package. It also can rarely mean the actual package file.

Build

A build is an ordering unit which contains a source package and multiple *build jobs*. There is also some meta information about the reason of the build (commit message, associated bugs). A build is what we are managing here. We don't bother about a single package file.

Build job or just: job

A build contains several jobs. One for each architecture the build is built for.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

PBS by example: A package

This is the kernel package.

The page contains some general information about the package and in the row below a list of all known builds.

At the bottom: Open bugs, some statistics.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

PBS by example: A build

A build can be created in two ways and thus they distinguish themselves in two categories.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

PBS by example: A build

A build can be created in two ways and thus they distinguish themselves in two categories.

- Release builds.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

PBS by example: A build

A build can be created in two ways and thus they distinguish themselves in two categories.

- Release builds.
- Scratch builds.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A release build

A release build is created out of a commit in a git repository.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A release build

A release build is created out of a commit in a git repository.

- It is automatically created from source.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A release build

A release build is created out of a commit in a git repository.

- It is automatically created from source.
- Always linked to a distribution.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A release build

A release build is created out of a commit in a git repository.

- It is automatically created from source.
- Always linked to a distribution.
- Goes its way through all repositories of the distribution (i.e. from testing to stable).

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A scratch build

In contrast to the release builds, a scratch build is created by an individual.

Command: *pakfire-client build beep/beep.nm*

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A scratch build

In contrast to the release builds, a scratch build is created by an individual.

Command: *pakfire-client build beep/beep.nm*

- Used to test if a packages does build on all architectures.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A scratch build

In contrast to the release builds, a scratch build is created by an individual.

Command: *pakfire-client build beep/beep.nm*

- Used to test if a packages does build on all architectures.
- Used to distribute experimental packages.

Introduction
The Pakfire Build Service
Learning by example
The end

Excursion: pakfire
More definitions
A package
A build

A scratch build

In contrast to the release builds, a scratch build is created by an individual.

Command: *pakfire-client build beep/beep.nm*

- Used to test if a packages does build on all architectures.
- Used to distribute experimental packages.

We will learn what exciting things we can do with them in a minute…

Step by step

Going step by step through the web user interface of the Pakfire
Build Service, you will get an impression about what it can do...

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Home / Packages / gcc

Package: gcc

Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

Release builds (2)

Build	Jobs	Repository	When
gcc-4.6.3-6.ip3	x86_64 i686 armv7hl armv5tel	IPFire 3 / testing	14 hours ago
gcc-4.6.3-5.ip3	x86_64 i686 armv7hl armv5tel	IPFire 3 / stable	Saturday at 6:57 pm

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Package: gcc

Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

At the top of the page, you will find the most important
metadata of the package

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Package: gcc
Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers		
Homepage	http://gcc.gnu.org/	**Description**	
License	GPLv3+ and GPLv2+ with exceptions	The gcc package contains the GNU Compiler Collection. You'll need this package in order to compile C code.	
Maintainer	★ Michael Tremer		

At the top of the page, you will find the most important
metadata of the package which is:
name and a short sentence what the package does (headline)

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Package: gcc
Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

At the top of the page, you will find the most important
metadata of the package which is:
name and a short sentence what the package does (headline),
a longer description

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Package: gcc
Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

At the top of the page, you will find the most important
metadata of the package which is:
name and a short sentence what the package does (headline),
a longer description, the home of the package (we call this
upstream)

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Package: gcc
Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

At the top of the page, you will find the most important
metadata of the package which is:
name and a short sentence what the package does (headline),
a longer description, the home of the package (we call this
upstream), license

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Package: gcc

Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

At the top of the page, you will find the most important
metadata of the package which is:
name and a short sentence what the package does (headline),
a longer description, the home of the package (we call this
upstream), license, and the maintainer within the IPFire project.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package

Release builds (2)			
Build	Jobs	Repository	When
gcc-4.6.3-6.ip3	x86_64 i686 armv7hl armv5tel	IPFire 3 / testing	14 hours ago
gcc-4.6.3-5.ip3	x86_64 i686 armv7hl armv5tel	IPFire 3 / stable	Saturday at 6:57 pm

At the bottom you will find a list of all builds for this package.

That are two release builds for this example, one in testing, the other already in stable state.

Let's have a closer look at them...

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build

Home / Packages / gcc / gcc-4.6.3-6.ip3

Build: gcc-4.6.3-6.ip3

Release build

Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

Update   Fixed bugs (0)

Repository

IPFire 3 - testing since 12:00 am

Commit	c6952ba - gcc: Fix a typo in the SSP patch.
Severity	Unspecified

Fix a typo in the SSP patch.

Push   Unpush

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build

Build: gcc-4.6.3-6.ip3

`Release build`

Various compilers (C, C++, Objective-C, Java, ...).

Group	Development/Compilers
Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer

Description
The gcc package contains the GNU Compiler Collection.
You'll need this package in order to compile C code.

The same metadata as we already know from the package page.

Additionally tagged as a release build or scratch build in the top right corner.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build

Further down, there is a block that shows us more information about the reason for this build and the current state.

Update	Fixed bugs (0)

Commit	c6952ba - gcc: Fix a typo in the SSP patch.
Severity	Unspecified

```
Fix a typo in the SSP patch.
```

Repository

IPFire 3 - testing since 12:00 am

Push   Unpush

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build

Further down, there is a block that shows us more information about the reason for this build and the current state.

This build for example was created by commit *c6952ba* and is in the testing repository.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build

At the bottom of a package is a short summary of the build jobs. You get an overview about how the build of the package goes.

Build jobs	Properties			
Arch	**State**	**Host**		**Duration**
x86_64	Finished	alecto.ipfire.org		1 h 29 m 42 s
i686	Finished	pomona.ipfire.org		2 h 48 m 41 s
armv7hl	Finished	winky.ipfire.org		6 h 36 m 37 s
armv5tel	Dispatching	dobby.ipfire.org		15 s

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build

By clicking on the *Properties* tab, you can see a couple of more meta information about this build.

Build jobs	Properties

Created	April 24, 2012 at 9:52 pm
Priority	Medium

Source package	gcc-4.6.3-6.ip3.src
Build dependencies	autogen, binutils>=2.21.51.0.8-1, dejagnu, elfutils-devel, expect, filesystem >= 002, flex, glibc-devel, gmp-devel, libffi-devel, libmpc-devel, mpfr-devel, texinfo, zlib-devel, cloog-ppl-devel and ppl-devel

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build: Comments & score

One major design goal of the PBS was the involvement of the community of testers into the development process.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build: Comments & score

One major design goal of the PBS was the involvement of the community of testers into the development process.

We achieved that by giving our users the opportunity to comment on builds and rate them like in this example:

Watchers: 0 people.

Comment   Score: 1

Michael Tremer - **2 minutes ago**
I tested this package on x86_64 where it still works.

+1

★ **Pakfire Build Service** - **14 hours ago**
This build was pushed to the repository 'testing'.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

We gain the following advantages:

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

We gain the following advantages:

- Users get go and grab a newer package if they are experiencing an issue with the current version.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

We gain the following advantages:

- Users get go and grab a newer package if they are experiencing an issue with the current version.
- If the package fixes their bug, they only need to type a few lines into the comment box and rate the build.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

We gain the following advantages:

- Users get go and grab a newer package if they are experiencing an issue with the current version.
- If the package fixes their bug, they only need to type a few lines into the comment box and rate the build.
- The developer gets immediate feedback about what he has created. This has been difficult in the past.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

We gain the following advantages:

- Users get go and grab a newer package if they are experiencing an issue with the current version.
- If the package fixes their bug, they only need to type a few lines into the comment box and rate the build.
- The developer gets immediate feedback about what he has created. This has been difficult in the past.
- If things don't go that well, it's the same game.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

We gain the following advantages:

- Users get go and grab a newer package if they are experiencing an issue with the current version.
- If the package fixes their bug, they only need to type a few lines into the comment box and rate the build.
- The developer gets immediate feedback about what he has created. This has been difficult in the past.
- If things don't go that well, it's the same game.

At the end of the day, it is much easier for the developer to decide if a package is broken or if the update is working well and worth being *pushed*.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

Who is allowed to comment
and vote?

Every registered user is
allowed to comment builds.

At the current point in time, it is
necessary to be enabled by an
admin to vote to assure a
certain quality.

Comment on kernel-3.2.12-3.ip3   ×

Comment   Since this update, my system does
                 crash after a short time.

                 -1

Vote   ○ Not tested.
                 ○ Works for me.
                 ◉ Does not work.

                                    Submit comment   Cancel

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A job

Build job: gcc-4.6.3-6.ip3.x86_64

Various compilers (C, C++, Objective-C, Java, ...).

| Job details | Time | Log files (1) |

State	finished	Schedule test build
Builder	alecto.ipfire.org	
Buildroot	117 packages	

On a job page, you will find a lot of information about the state and progress of a build job for a certain architecture.

NOTE: The interface may still change a bit.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

You have access to the build log files which contain the whole build process and are very useful for debugging.

Job details	Time	Log files (1)

Filename	Size	
build.x86_64.1.log	4M	⏻ Download

They are easy to download or can be directly viewed in the web browser.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

Another very important feature is the so called *buildroot*.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

Another very important feature is the so called *buildroot*.

If there is a package with a serious bug which affected other packages that have been built with that package, we need to know which packages are infected.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

Another very important feature is the so called *buildroot*.

If there is a package with a serious bug which affected other packages that have been built with that package, we need to know which packages are infected.

The buildroot gives us a list of all packages that were extracted to the buildroot at this build job.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

Another very important feature is the so called *buildroot*.

If there is a package with a serious bug which affected other packages that have been built with that package, we need to know which packages are infected.

The buildroot gives us a list of all packages that were extracted to the buildroot at this build job.

Another application: If a build fails, we are able to create the exactly same buildroot and reproduce the complete build process.

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

Job buildroot: gcc-4.6.3-6.ip3.x86_64

This is the buildroot of build job gcc-4.6.3-6.ip3.x86_64.

The packages listed below were used for the build.

- audit-libs-2.2-2.ip3.x86_64
- autogen-5.14-1.ip3.x86_64
- autogen-libs-5.14-1.ip3.x86_64
- bash-4.2-9.ip3.x86_64
- beecrypt-4.2.1-2.ip3.x86_64
- binutils-2.22-3.ip3.x86_64
- bzip2-1.0.6-6.ip3.x86_64
- ca-certificates-2012.81-1.ip3.noarch
- ccache-3.1.7-1.ip3.x86_64
- cloog-ppl-0.15.11-1.ip3.x86_64
- cloog-ppl-devel-0.15.11-1.ip3.x86_64
- coreutils-8.16-3.ip3.x86_64
- cpio-2.11-2.ip3.x86_64
- cpp-4.6.3-5.ip3.x86_64
- cracklib-2.8.18-2.ip3.x86_64
- cracklib-dicts-2.8.18-2.ip3.x86_64
- cyrus-sasl-libs-2.1.25-2.ip3.x86_64
- db4-1:4.8.30-5.ip3.x86_64
- delagpu-1.4.4-2.ip3.noarch

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A build job also owns the package files that came out of the build process.

Package files (13)

Name	Version	Arch	Size	
cpp	4.6.3-6.ip3	x86_64	4M	ⓘ Download
gcc	4.6.3-6.ip3	x86_64	7M	ⓘ Download
gcc-c++	4.6.3-6.ip3	x86_64	5M	ⓘ Download
gcc-debuginfo	4.6.3-6.ip3	x86_64	27M	ⓘ Download
gcc-plugin-devel	4.6.3-6.ip3	x86_64	560k	ⓘ Download
libgcc	4.6.3-6.ip3	x86_64	120k	ⓘ Download
libgomp	4.6.3-6.ip3	x86_64	40k	ⓘ Download
libmudflap	4.6.3-6.ip3	x86_64	80k	ⓘ Download
libmudflap-devel	4.6.3-6.ip3	x86_64	20k	ⓘ Download
libquadmath	4.6.3-6.ip3	x86_64	130k	ⓘ Download

Let's click on one of these...

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

A package file

Again, we have a lot of meta information which explains itself.

Package: libgcc-4.6.3-6.ip3.x86_64

GCC shared support library.

This package contains GCC shared support library which is needed e.g. for exception handling support.

Homepage	http://gcc.gnu.org/
License	GPLv3+ and GPLv2+ with exceptions
Maintainer	★ Michael Tremer
Build host	alecto.ipfire.org
Build time	April 25, 2012 at 12:04 am UTC

Source package	gcc-4.6.3-6.ip3.src
Build	gcc-4.6.3-6.ip3
Job	gcc-4.6.3-6.ip3.x86_64
Size	120k (120k when installed)

⏻ Download

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

You may view the dependencies of a package online…

Dependencies

Provides   Requires

- libgcc_s.so.1()(64bit)
- libgcc_s.so.1(GCC_3.0)(64bit)
- libgcc_s.so.1(GCC_3.3)(64bit)
- libgcc_s.so.1(GCC_3.3.1)(64bit)
- libgcc_s.so.1(GCC_3.4)(64bit)
- libgcc_s.so.1(GCC_3.4.2)(64bit)
- libgcc_s.so.1(GCC_3.4.4)(64bit)
- libgcc_s.so.1(GCC_4.0.0)(64bit)
- libgcc_s.so.1(GCC_4.2.0)(64bit)
- libgcc_s.so.1(GCC_4.3.0)(64bit)
- libgcc_s.so.1(libgcc_s.so.1)(64bit)

Introduction
The Pakfire Build Service
Learning by example
The end

Step by step
A package
A build
A job
A package file

…as well as the list of files in the package (with permissions and user) which is both very useful for debugging.

Files

drwxr-xr-x	root:root	0B	/usr/	V D
drwxr-xr-x	root:root	0B	/usr/lib64/	V D
-rw-r--r--	root:root	413k	/usr/lib64/libgcc_s.so.1	V D

Conclusion

In conclusion, the Pakfire Build Service has increased our productivity enormously, because it is so much faster in bringing out package updates.

Conclusion

In conclusion, the Pakfire Build Service has increased our productivity enormously, because it is so much faster in bringing out package updates.

Bugs get fixed very fast and do not stop you working on something.

Conclusion

In conclusion, the Pakfire Build Service has increased our productivity enormously, because it is so much faster in bringing out package updates.

Bugs get fixed very fast and do not stop you working on something.

Developers are able to test each others changes very easily and provide feedback at a central place. So are users.

Conclusion

In conclusion, the Pakfire Build Service has increased our productivity enormously, because it is so much faster in bringing out package updates.

Bugs get fixed very fast and do not stop you working on something.

Developers are able to test each others changes very easily and provide feedback at a central place. So are users.

The integration with the bugtracker (Bugzilla) closes bugs and makes sure that the reporter of a bug is informed about a fix and encouraged to test it.

Altogether, we are very proud of what we have done.

Altogether, we are very proud of what we have done.

The Pakfire Build Service makes development fun. It helps to achieve higher goals.

Altogether, we are very proud of what we have done.

The Pakfire Build Service makes development fun. It helps to achieve higher goals.

We don't consider the service fully done, because we have so many ideas on our minds…

THE END

Now go out and show what you have learned today at
https://pakfire.ipfire.org