



## Open Source Software ENSI Club



# Open Source Magazine

Arch Linux | WebRTC | FirefoxOS

Conventus

--Technologie--

Raspberry pi:  
GameBoy Pocket

--Security--

Who will Win  
The security war ?

-Programmation-

JAVA 8  
8 reasons to start using it

# Open Source Software ENSI Club

It is with great pleasure that we release the 3rd Number of the OSSEC magazine entitled "Conventus" ..

As always , the magazine tries to bring you a variety of information and news concerning open source , brought to you by various article writers coming together from different places and engineering schools .. That's the spirit : open source unites us all :)

The OSSEC team would like to take this opportunity to thank all the contributors for their great work and we hope we're getting closer to our readers's expectations ..

We were pleased to receive your feedback on the last issue and we would like to hear from you this time as well , so remember that your opinion matters to us and send us an e-mail !

Have a pleasant reading !



# Summary

- Open source  
Why to open source your startup
- C-doku
- PostgreSQL
- Firefox OS
- Arch Linux  
Distribution d'élites?
- WebRTC Framework
- TrueCrypt

P03

P08

P09

P11

P13

P21

P29

Programmation:

— Java8

Security:

— How will win the security war

Technology:

— Raspberry Pi

Open tips

Entertainment:

— Quiz!!

Events:

— Forum / Hackathon

About us / Contact / Team

P05

P19

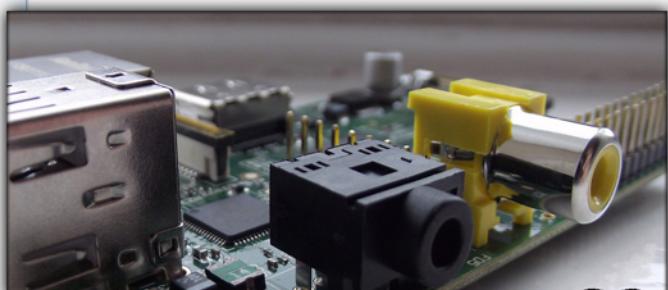
P25

P28

P31

P32

P33





# Open Source

## Why to open source your start-up ?

Being an entrepreneur by nature means that you are comfortable with being challenged and are willing to take risks for the best interest of your company ..

Now , you finally managed to launch a start-up , what would make you consider open sourcing your start-up ?

I'm here to try to show you what a great idea it is by answering some of the questions you might ask ..

### Ok so why would I even consider open sourcing my start-up ?

Well, the advantages are numerous , first of all , by putting your code out there , many contributors will offer to help, even simple users of your software can give their opinion and signal the bugs they face .. Look at the speed in which your product will grow compared to staying limited to a small team .. Besides , open-source products are proven to be more reliable than proprietary ones where you're asked to trust the good judgment of the developers ..

### Don't closed source companies take the open source code and then add to it ?

It's completely legitimate to worry about your code , this field is competitive , but hopefully the GPL



There's a saying that goes : "He who fears something gives it power over him." , and that's exactly what happened with those companies , a start-up called CodeCombat went totally open-source about a year ago , it's a project that aims to teach people to code through a game .. These people asked their advisors before going open-source , their lawyers , investors and developers and the funny thing is that each of them didn't mind but thought that the other part might or maybe there would be a reason why they shouldn't ... that wasn't the case , the project went open-source and thanks to that grew in a better way than it would have if just the tiny group of developers worked on it .. They talked about their experience with open sourcing their project in a blog : <http://blog.codecombat.com/why-you-should-open-source-your-startup>

You see ? There isn't really a good reason why you shouldn't open-source your start-up , it's just your fear that is stopping you from doing so ..

Alright so to sum up what we just said , life's adventure is in taking risks and yes open-sourcing your start-up might be a risk at the beginning but its benefits are far more than the drawbacks , experts say so , the best example is **Mark Shuttleworth** who said : "There are many examples of companies and countries that have improved their competitiveness and efficiency by adopting open source strategies. The creation of skills through all levels is of fundamental importance to both companies and countries."

Are you considering open sourcing your start-up yet ?

license protects your rights as anyone who uses your code cannot deny others from the rights he had , meaning if closed source companies used your code , they'd be obliged to open source theirs .. Good deal right ?

### Will the contributors help be significant ?

Putting your project on GitHub , the contributions can be tiny, because they come from new beginners either to open source or to programming . But it's not a waste of time because as long as there are contributions there's progress ..That's the spirit of open source : help and ask for help .. everything becomes easier when shared...

### If open sourcing is so interesting , why are most companies closed source ?

Lamia Ben Neticha



# 8 reasons to start using java 8 if you haven't already!

You may be familiar with the small updates java releases every 2 years approximately where you just have to update your JDK without any major change in your coding routine.

Well the long-awaited java 8 release is different. On March 2014 this release came up with the biggest upgrade to the java programming modal ever. The new features and new syntax made it clear that this upgrade was going to have an impact on the programmer's coding habits, but is that good news or bad news?

I'll let you judge by yourself.

## 1. We can't talk about java 8 without mentioning the project lambda:

Lambda expression is definitely the most important feature to enter the java language since generics in java 5. It represents the big shift into the java programming modal allowing a functional coding style that shortens your code while making it more readable.

## 2. Extension methods (this is definitely one of my favorite features):

-More explicitly called default methods for interfaces (which sounds pretty unusual, I agree), the extension methods are non-abstract methods implemented (yes implemented, I'm not kidding you guys) inside an interface.

-All you have to do to make this magical extension methods work is by adding the default keyword before the method.

-This way concrete classes only have to implement the good old abstract methods without worrying about the default methods that can be used out of the box.

-Default methods are really interesting in a way they enable us to add a new default method to a JDK interface without breaking the implementation isn't that wonderful?

-I invite you to dig a little more in this amazing feature, the `java.lang.Iterable` is a really good example since the `forEach` method is a default one.

PROGRAMMING

A TOOL FOR CONVERSATION





# HAMMER: ING CAFFEINE INTO CODE

### 3. Functional interfaces or the way to use lambda expressions:

- Each lambda is specific to a type given by an interface and you can guess now what the so called functional interfaces are.
- Functional interfaces are simply interfaces with one and only one abstract method so that each lambda of that type would be matched to this abstract method.
- In a functional interface you could add as many default methods as you want since they are non-abstract methods.
- Also if you're feeling the need to be more consistent when using Functional interfaces you could add the annotation `@FunctionalInterface` before your interface. This way your compiler won't be happy when you try to add a second abstract methods and he will definitely make you understand that.

### 4. Methods and constructors references:

- Those of you who are all about less lines and neater code are going to love this feature.
- With the methods and constructor references java 8 is digging deeper into the simplification ground by making the minimum number of lines a priority.
- Java 8 enable us to pass method or constructor references via a simple `::` keyword

### 5. Built-in functional interfaces:

- In addition to the existing built-in functions that were extended using `@Functional Interface` to enable lambda expression, the java 8 API came with a whole bunch of new functional interfaces to make your day to day coding even smoother.

#### Predicates

Predicates are Boolean valued functions that takes only one argument, also the interface contains a good number of logical terms.

Since a code snippet is worth a thousand words here is the predicate interface:

```
package java.util.function;

@FunctionalInterface
public interface Predicate<T extends Object>{
    public boolean test(T t);

    public default Predicate<T> and(Predicate<? super T> prdct){
        //compiled code
    }

    public default Predicate<T> negate() {
        //compiled code
    }

    public default Predicate<T> or(Predicate<? super T> prdct) {
        //compiled code
    }

    public static <T extends Object> Predicate<T> isEqual(object o) {
        //compiled code
    }
}
```

## Functions

The one thing you have to retain about functions is that they take one argument and produce one result. Another good thing to know is that we can use a default method like andThen to link two functions. To make all crystal clear here's the lovely snippet:

```
package java.util.function;  
  
@FunctionalInterface  
public interface Function<T extends Object, R extends Object> {  
    public R apply(T t);  
    public default <V extends Object> Function<V, R> compose(Function<? super V, ? extends T> fnctn) {  
        //compiled code  
    }  
    public default <V extends Object> Function<T, V> andThen(Function<? super R, ? extends V> fnctn) {  
        //compiled code  
    }  
    public static <T extends Object> Function<T, T> identity() {  
        //compiled code  
    }  
}
```

## Suppliers

The name intimates it explicitly, suppliers don't accept arguments and they happen to produce a result.

## Consumers

Consumers on the other hand take one and only one input to perform a certain operation on.

```
package java.util.function;  
  
@FunctionalInterface  
public interface Supplier<T extends Object> {  
    public T get();  
}
```

```
package java.util.function;  
  
@FunctionalInterface  
public interface Consumer<T extends Object> {  
    public void accept(T t);  
    public default Consumer<T> andThen(Consumer<? super T> cnsmr) {  
        //compiled code  
    }  
}
```

## 6. Streams and ParallelStreams:

java.util.stream is a sequence of elements and on these elements we can perform numerous operations.

There are two types of operations terminal ones and intermediate ones (return a stream so that we can call another method on it)

Streams are created on java.util.collection like list to be explicit.

Streams can be executed in a sequential way or parallel way using multiple threads which of course is faster.

Let's say we have a list of words called words, we could use a great number of stream operations like:

### Filter:

This is one of the most used intermediate stream operations

### Sorted:

Another intermediate stream operation that could be very useful

Since sorted is an intermediate stream operation here is an example using sorted() and filter()

There are also other really interesting operations like: map, match, count and reduce.

## 7. The date API:

You may have noticed that the JDK classes Date and Calendar hadn't had that good of a design, they also tend to have numerous bugs, so the new date and time API is a widely expected feature in java 8.

This new API has a lot to offer.

-A Clock to access the current date and time

```
words.  
.stream()  
.sorted()  
.filter((s) -> s.startsWith("a"))  
.forEach(System.out::println);
```

-An Instant class to create legacy java.util.Date objects

-Timezones

-LocalTime

-LocalDate

-LocalDateTime



## 8. Annotations:

The great thing about annotation in java 8 is that they are repeatable.

But for an annotation to be repeatable we must declare it as so.

The repeatable annotation can be particularly useful for scheduling a task for two different dates.

With the @Schedule(dayOfWeek="", hour="")

From these 8 new features, java 8 is all about shorter, more readable code and a smoother day to day coding experience. This article is only a taste of the update so no more waiting!

Turn off that movie you're watching and log out of whatever social media you're scrolling back and forth on right now and go get started!

You don't want to make java 8 wait longer, would you?

# C-doku

## Introduction :

First year students of the national school of computer science of Tunisia are required to develop an application in C language. We introduce to you hereby a project among them that was ... open source which is C-doku



## What is C-doku ?

C-doku is a standard SUDOKU puzzle developed entirely in C language. It offers two modes of execution => CLI and GUI. For the moment you decide which mode on compile time.

It uses SDL1.2 for the graphics. It is developed to make it easy to maintain and improve its functionalities, thus its code may be used as a base for other projects alike . It is easy to adapt the code to make it work for grids of different sizes for example.

## How to get the game?

In order to get the game, you need to have git installed along with libSDL1.2-dev, libSDL-image1.2-dev, libSDL-ttf1.2-dev and gcc.

For Debian based systems, execute the following commands

```
# apt-get install git libSDL1.2-dev libSDL-image1.2-dev libSDL-ttf1.2-dev
```

```
$ git clone https://github.com/embedMaster/sudoku.git  
c-doku && cd c-doku
```

For command line interface :

```
$ make console
```

For Graphical User Interface :

```
$ make sdl
```

You can build both via :

```
$ make
```

Windows users will find an installer (sudoku\_setup.exe)

How to contribute ?

Fork the repository, modify it, submit a package with name, description, URL, and optional screenshots. Don't forget to submit a pull request.

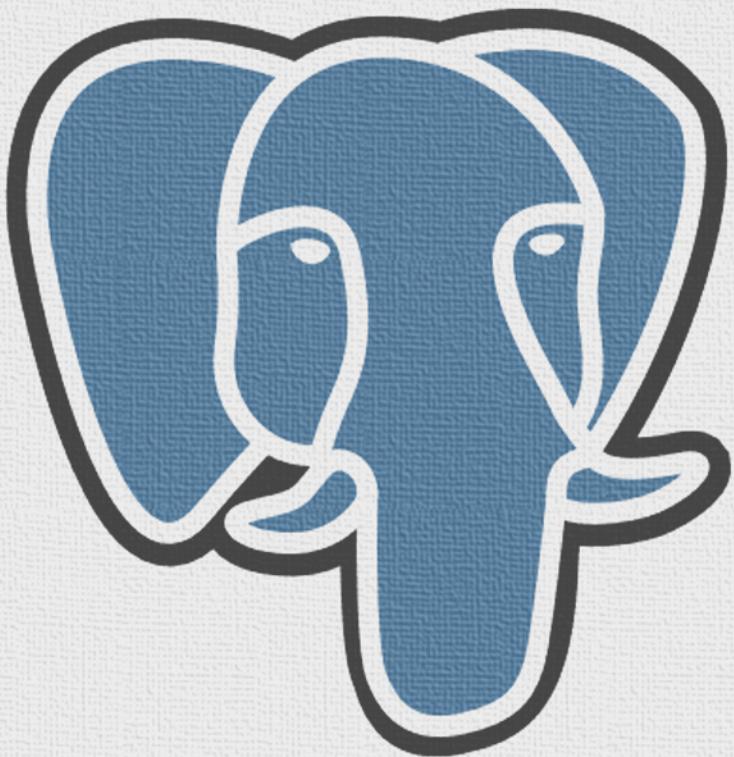
Suggested possible improvements:

Writing an automatic solver system

Writing a grid generator

Using sound effects...





# PostgreSQL

## Scaling up PostgreSQL

I am expecting 1000+ hits on my PostgreSQL server and I doubt my standalone database will be able to handle it ? Seriously, "replication and connection pooling" are indeed the terms you want.

There are many approaches available to scale PostgreSQL beyond running on a single server (standalone node). An outline of the terminology and basic technologies involved is at High Availability and Load Balancing. Deploying PostgreSQL in a high-demand environment requires reliability and scalability and fortunately PostgreSQL's ecosystem offers a set of tools you need to build out a robust database system.

Typically, a fault-tolerant system does not just deploy a single server ! Rather a PostgreSQL cluster is deployed. A cluster consists of a master (sometimes several masters) PostgreSQL server, and one or more replication slaves. Writes are send to the master, and all slave servers are available to serve read requests. Often, requests are dispatched via a loadbalancing tool/algorithm. If the master fails, one slave can be promoted to be the new master. This is called a failover process. Once the old master is up again, a fallback process is launched in order to reattach the node to the cluster again.

**Replication :** For database systems, replication is the process of sharing transactional data to ensure consistency between redundant database nodes, typically slaves. This improves fault tolerance, which leads to better reliability of the overall system. Replications systems for databases can also be called distributed databases. Basically there are 3 types of replication in PostgreSQL i.e Warm, hot standby and Streaming Replication. Replication is often coupled with load balancing to improve read performance. Some replication solutions, like Pgpool, have an integrated load balancer, which knows about the underlying system.

**Connection pooling :** connection pool is a group of database connections sitting around that are waiting to be handed out and used. This means when a request comes in a connection is already there whether in your framework or some other pooling process, and then given to your application for that specific request or transaction. In contrast, without any connection pooling the application will have to reach out to the database, for each query, to establish a connection. While in the most basic sense you may thinking connecting to a database is quick, often theres some overhead.

Sometimes, a full-blown you don't really know what to use. A good question is "Which shall I deploy with PostgreSQL ?"

- If you want to eliminate the overhead of creating and tearing down new connections (pooling), particularly over a network interface, PgBouncer is a better choice
- Use Pgpool if you want to leverage your cluster for load balancing and failover.
- To set up and monitor a cluster replication, Repmgr is a good solution.

Finally , if you're lazy and you find configuring and orchestrating a hign available cluster complicated, use Puppet with repmgr-puppet module. It'll do all the stuff for you.

Ahmed Bessifi

#### Some references (for more infos) :

<https://github.com/abessifi/repmgr-puppet>

<https://github.com/abessifi/pgpool-online-recovery>

<https://wiki.postgresql.org/wiki/PgBouncer>

<http://www.postgresql.org/docs/9.1/static/warm-standby.html#SYNCHRONOUS-REPLICATION>



# Firefox OS

## the new kid in the neighborhood

The word "Open Source" is fair enough to describe Firefox OS, Mozilla's new operating system for mobile. But in fact, what is it? How is it different from other mobile operating systems? And what goals Mozilla aims to accomplish with it?

Mozilla, the non-profit organization, is a global open source community of technologists, thinkers and builders working together to keep the Internet alive and accessible, that cares about web technologies and from here comes its vision of the web as a secure, transparent, free and open-to-everyone space where everyone can and is allowed to get the information they are looking for, without any barriers, or any concerns that someone is watching them.

So until now, you basically know that Firefox OS is an open source mobile operating system maintained by Mozilla. Let me get you into the details by answering the common questions anyone would ask.

**Q:** What platforms Firefox OS is targeting?

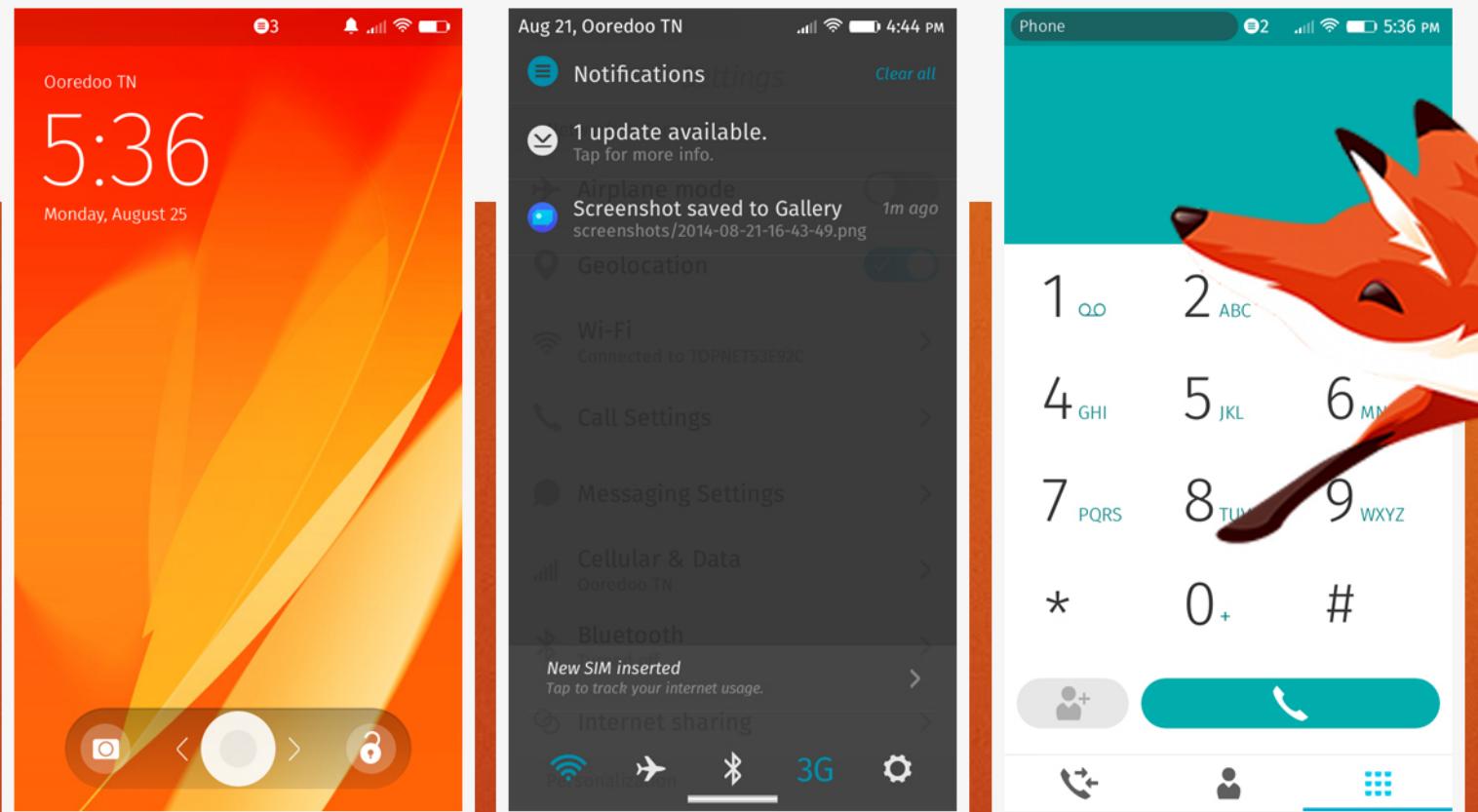
**A:** Firefox OS aims to be on mobile devices, basically low-end smartphones, and tablets. In the future you will also see Smart watches and TVs running Firefox OS.

**Q:** How is that different from Android in terms of Open Source software?

**A:** Android isn't quite open source, to be honest, because you don't get to contribute to the development of any version UNTIL Google releases it to the public, so basically only Google employees work on Android's features and fix its bugs.

However Firefox OS is a different story, it has been fully open source and accessible to everyone from Day 1, since it was just an empty repository. You can get to know what features implemented, what is currently being worked on, what's planned for the Future of Firefox OS.

**Q:** What goals Firefox OS is aiming to accomplish?



**A:** Firefox OS is NOT aiming to compete with Android neither iOS or any other Mobile OS, Firefox OS' mission is to bring the "Smart" mentality (if that's the right word) to developing countries, how? Because smart devices for Firefox OS (including Smartphones, Smart Watches and Smart TVs) will be sold at a very low price, e.g. We have a Firefox OS smartphone that will be sold to all over the world with a 25\$ price tag. Another thing, Firefox OS is built using web technologies, as I mentioned before, Mozilla is known to defend open web standards, so built on top of web technologies here means that all the Applications and Files and codes in the System (more specifically the user interface, UI) are and can be built using HTML5.

**Q:** I tried Firefox OS and some features are still missing, any plans for future implementation?

**A:** The project started back in 2011 and still in development, as it's the case for Android which was started by Google in 2007, we (the Middle East and North Africa) didn't start hearing about it and its potential until 2011 - 2012 which leaves a great gap here, that gap was filled by Google testing and improving its OS until it became world-wide available.

To make the long story short: Firefox OS aims to bring the power of the web to the next billion human, those who couldn't afford 600\$ to waste on an iPhone.

---

Ahmed Nefzaoui



Written by Ahmed Nefzaoui  
with help from Manel Rhaiem



# Arch Linux

## Distribution d'élites ?

Arch linux, distribution mûrie inspirée de Crux Linux et créée par le jeune Judd Vinet, un jeune développeur open source. Basé sur le principe Kiss, arch linux avec sa philosophie attirante est supposé être l'un des plus simples et légères distributions.

"Un designer sait qu'il a atteint la perfection non pas quand il n'y a plus rien à ajouter, mais quand il n'y a plus rien à enlever." — Antoine De Saint-Exupery. Très optimisé et Dépourvu d'une interface graphique, un certain niveau de connaissances est nécessaire pour l'installer et le configurer en fonction des besoins.

La licence GPL et la communauté d'utilisateurs et de développeurs fidèles d'Arch Linux, composée 100% de bénévoles, rend arch une distribution libre par excellence.



### Arch phylosophy! — KISS «Keep it simple, Stupid» —

Le projet Arch vise la simplicité de conception en évitant toute complexité non nécessaire, il suit le principe KISS « Garde ça simple, stupide » ou encore « Ne complique pas les choses ». Pourquoi alourdir son système avec des applications et des fonctionnalités inutiles ? C'est sur cette idée que Arch linux est fondé, un environnement minimal qui éloigne tout paquet inutile ou indésirable, c'est à l'utilisateur d'administrer son système afin de créer son propre environnement personnalisé selon ses besoins. Certainement ce n'est pas la simplicité

attendue : une interface graphique pour gérer tout, une installation rapide avec une configuration automatique, mais une complexité technique réduite qui permet de tout gérer facilement suivant une conception simple.

### — DIY «Do it yourself» —

"Arch est ce que vous en faites !" Affirme -Judd Vinet-. Étant un utilisateur de Arch vous êtes l'administrateur, c'est à vous de tout configurer pour construire un



système adapté à vos propres choix. La plupart des distributions cachent les bits de configuration derrière des programmes graphiques guidés et simples à manipuler, ce n'est pas le cas d'Arch linux : toutes les possibilités d'accès aux fonctionnalités sont permises.

## Les caractéristiques d'Arch

### «Rolling Release»

Contrairement à Ubuntu, Mint et Opensuse, Arch linux est un système en publication "roulante" ou "continue" «Rolling Realease System», c'est une approche qui consiste à présenter un flux quotidien constant de mises à jour sur le serveur, accessible par les

utilisateurs après une période d'essai déterminée par la communauté. Arch linux, utilisant ce principe, vous fait gagner le temps de réinstaller tout votre système pour avoir les dernières versions, `pacman -Syu` vous suffit pour obtenir les mises à jour relatives à la dernière version disponible. Le principe rolling release vous permet aussi d'obtenir les dernières versions d'applications dès leur diffusion, vous obtenez la dernière version de gnome, kde et xfce avant les autres distributions où vous êtes pluparement obligé de réinstaller entièrement le système pour des problèmes de compatibilité.

### «Pacman»

Combinant la fiabilité et la facilité, Pacman



(le gestionnaire de paquets officiel de la distribution Arch Linux) est l'un des caractéristiques distinctives d'Arch. En effet, il offre un maximum de contrôle pour garder votre système à jour tout en utilisant des commandes simples. Il facilite également, l'empaquetage des applications par rapport à "deb" et "rpm".

La configuration de Pacman se fait à partir du fichier « /etc/pacman.conf ».

Citons quelques commandes utiles :

**pacman -Q** - affichage des paquets installés.  
**pacman -S** - téléchargement du paquet  
**pacman -R** - suppression du paquet  
**pacman -Syu** - mise à jour de tous les paquets

**pacman -Sg** -affichage des groupes de paquets disponibles

**pacman -Si** -Donne les informations du paquet en argument.

Pour avoir la liste complète des commandes qu'offre pacman ,vous pouvez consulter <https://wiki.archlinux.fr/Pacman>

### Le depot «AUR»

Les dépôts stables d'Arch comportent le dépôt "core" : le plus minimal , inclus seulement les paquets nécessaires au fonctionnement du système, le dépôt "extra" : entretenu par les développeurs d'arch ainsi que d'autres paquets optionnels par rapport à "core", le dépôt "multilib" pour l'architecture x86\_64, et finalement le dépôt "community"

qui contient les paquets d'AUR qui ont gagné la confiance des utilisateurs . Alors c'est quoi AUR ?

<<Arch User Repository>> est un endroit dédié aux utilisateurs et qui compte plus que 44000 paquets, c'est un dépôt riche qui demande un peu de prudence car vous risquez de tomber sur des paquets non fiables ou corrompus!. Le groupe d'arch est strict dans la qualification d'un paquet standard, car il donne plus d'importance à la qualité des paquets qu'à leur quantité, c'est sur ce principe que AUR est fondé , on présente une chance aux utilisateurs pour jouer leurs tours afin d'obtenir la confiance de la

communauté.

Afin d'installer un paquet d'AUR:

- téléchargement de l'archive relative au paquet.

- extraction

- `makepkg -s`

- utiliser "pacman" pour installer le paquet créé.

Ou encore utiliser "yaourt", disponible dans les dépôts AUR, pour installer facilement:

- yaourt nomDuPaquet

## Distributions basées sur Arch

Le succès d'Arch Linux a permis l'apparition de plusieurs distributions basées sur son système léger et minimal (même pas une interface graphique) qui offre plein de choix aux développeurs.

### Manjaro

Vous avez certainement entendu parler de Manjaro, une jeune distribution stable fondée et développée par "Roland Singer", "Guillaume Benoit", "Philip Müller" sous le principe d'être fidèle à Arch Linux mais en facilitant l'utilisation aux débutants. Manjaro offre officiellement deux environnements graphiques , Xfce et Openbox disponibles tous les deux en version 32-bits et 64-bits dans le site officiel "<http://manjaro.org/get-manjaro/>" , possédant par défaut une gamme des logiciels les plus utilisés comme Firefox, VLC et LibreOffice. Il supporte également les environnements de bureau Kde, Gnome, Mate .... il ne se limite pas aux dépôts d'Arch mais utilise de nouveaux dépôts propres à lui. Il est à noter que Manjaro est classé en 10ème position pour les 12 derniers mois dans le site DistroWatch.

<http://manjaro.org/>

<<Manjaro Wiki: [https://wiki.manjaro.org/index.php?title=Main\\_Page](https://wiki.manjaro.org/index.php?title=Main_Page)>>

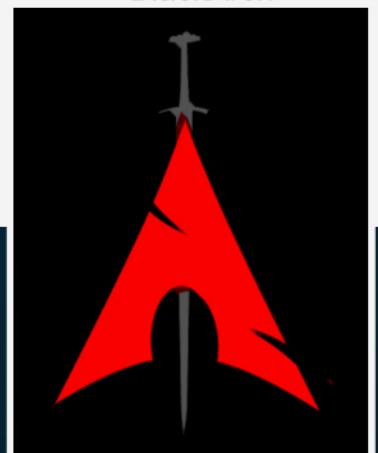
ArchBang



Manjaro



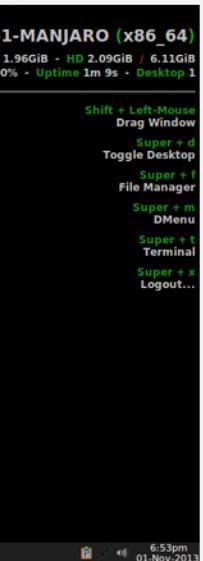
BlackArch





## 2012 - Arch Linux

Arch Linux a gagné la "Free Software Award" pour la distribution la plus favorable au consommateur Linux. Aaron Griffon a reçu le prix de Richard M. Stallman et Linus Torvalds lors de la conférence LibrePlanet à l'Université du Massachusetts, à Boston.



**ArchBang**, une autre jeune distribution basée sur Arch et utilisant le bureau OpenBox, elle est rapide stable et simple en utilisation cependant elle possède un support de langue limité et une collection de logiciels de base beaucoup moins riche que celle de Manjaro.

<<[>>](http://wiki.archbang.org/index.php?title=Main_Page)

**BlackArch**, une distribution de pentest pour les professionnels en sécurité et les testeurs de pénétration, elle comporte plus que 1000 outils prêts à être utilisés.

<<<http://www.blackarch.org/>>>

<<[liste des outils de Blackarch](http://www.blackarch.org/tools.html): <http://www.blackarch.org/tools.html>>>

**BBQLinux**, une distribution basée sur arch, utilise les mêmes paquets et la même philosophie mais dédiée pour les développeurs Android. elle utilise Mate comme environnement de bureau par défaut.

<<<http://bbqlinux.org/>>>

il existe plein d'autres distributions basées sur Arch : BlueStar Linux, chakra ....



## Installation

Contrairement à la plupart des autres distributions, Arch n'utilise pas une interface graphique guidée pour l'installation. En fait, tout est géré à partir de la console. C'est ce qui rend la tâche un peu pénible et non dédiée aux débutants dans le monde des distributions Linux. Ne vous inquiétez pas trop, car vous n'aurez à effectuer l'installation qu'une seule fois puis vous allez bénéficier d'un système à jour grâce à pacman -Syu que nous venons de présenter. Le guide fourni par la communauté présente presque tous les détails et les astuces pour l'installation. On va essayer de simplifier la procédure et faciliter la tâche :

### Prérequis

- 1- connaissance des commandes Shell de base
- 2- Maîtrise d'un éditeur de texte en console: vi ou nano (pour débutants)

### I- Préparation de l'image sur un disque USB

nb: 1- Vérifier bien que le disque n'est pas monté  
2- Remplacer sdx par votre disque (exemple : /dev/sdb)  
(tapez #fdisk -l pour lister tous les disques et connaître leurs lettres )

commande :

```
# dd bs=4M if=/chemin/de/l'iso/archlinux*.iso of=/dev/sdx && sync
```

### Remarque

après le démarrage de l'iso, le clavier est configuré en anglais (qwerty), tapez la cmd: "loadkeys fr" pour le changer en français (azerty)

## II- Partitionnement du disque

Il est vivement conseillé d'utiliser un logiciel graphique tel que Gparted pour le partitionnement de votre disque dur. Il est beaucoup plus pratique et permet d'éviter des erreurs irréversibles. Vous pouvez l'utiliser à partir d'un "Live CD" exemple de partitionnement : /dev/sda1 : /  
/dev/sda2 : swap

## formatage des partitions et création du swap :

```
$ mkfs.ext4 /dev/sda1  
$ mkswap /dev/sda2
```

## monter les partitions :      activez le swap :

```
$ mount /dev/sda1 /mnt      $ swapon /dev/sda2
```

## III- Connexion réseau

-utiliser : \$ wifi-menu pour se connecter a un wifi (il faut que votre pc détecte la carte réseau wifi)

## IV- Installation du système de base

```
# pacstrap /mnt base base-devel
```

## V- Démarrage

```
installez grub: $ pacstrap /mnt grub
```



## VI- Paquets supplémentaires

1) Installez les paquets suivants pour être capable d'établir une connexion internet une fois connecté à votre système après la fin de l'installation

```
# arch-chroot /mnt /bin/bash  
# pacman -S wireless_tools, wpa_supplicant, dialog
```

2) Si vous désirez avoir un environnement graphique, vous devez installer le serveur graphique Xorg ainsi que les pilotes de votre carte graphique.

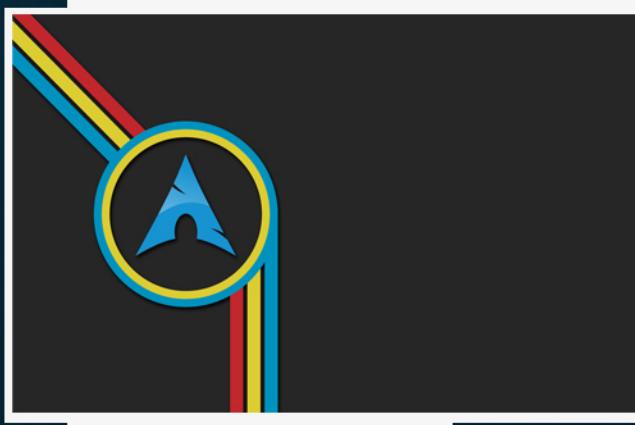
Pour plus de détails , vous pouvez visiter le guide officiel d'installation d'Arch linux:

En Français : [https://wiki.archlinux.fr/Installation#Installation\\_de\\_l.27ISO\\_sur\\_une\\_cl.C3.A9\\_USB](https://wiki.archlinux.fr/Installation#Installation_de_l.27ISO_sur_une_cl.C3.A9_USB)

En Anglais (plus complet et détaillé) : [https://wiki.archlinux.org/index.php/Beginners%27\\_guide](https://wiki.archlinux.org/index.php/Beginners%27_guide)

Pour finir, Arch linux est une distribution très puissante qui vous permet de mieux comprendre votre Système en vous fournissant le contrôle nécessaire pour le configurer convenablement selon vos besoins. L'installation non équipée d'une interface graphique et la configuration plupartement manuelle rendent Arch une distribution non dédiée aux débutants de Linux. Alors, C'est limité aux élites ? pas nécessairement; l'Arch Wiki est équipé de tous les guides et les ressources dont vous aurez besoin en tant qu'utilisateur Linux pour gérer votre environnement.

Haithem Abdelli





**Linux  
Professional  
Institute**



## Formation Prévue par L'ossec

lpi 101	06 - 10 -2014	16h.30
	08 - 10 -2014	14h.00
	09 - 10 -2014	16h.30
	11 - 10 -2014	14h.00
lpi 102	20 - 10 -2014	16h.30
	22 - 10 -2014	14h.00
	23 - 10 -2014	16h.30
	25 - 10 -2014	14h.00



**Formation 100% gratuite**

**Certification à 210dt au lieu de 300dt**

| [contact@ossec.tn](mailto:contact@ossec.tn)

# who will win the security war ?

## Open source softwares or commercials softwares !

Many people including developers still believe the myth that says open source is more vulnerable than commercial applications because of its source code availability to the public access.

It seems a little bit logical to think that when bad people look at the code and discover that the code contributer didn't drink enough coffee and let some input unverified, they will hunt down every person who is currently using the program on his machine, well this is not true because in general, people who find bugs and often vulnerabilities don't look at the source code, they are using specific programs called debuggers or disassemblers to trace the binary or the executable, so on the opposite this feature guarantees the most powerful benefit of open source world, its security !

Yes you are reading it right the open source security, every security flaw or vulnerability can be undertaken in few days if not in few hours, why ?, simply because of the great number of developers working on such a project.

According to "ohloh.net" website which is a data

analysis company for open source projects there are about 664.740 open source projects in the world that count 30.229.414.348 lines of code with 3.485.096 open source contributors ready to fix issues like security flaws . Even companies like Google or Microsoft can't afford to hire such a huge number ..

"So what ? Companies like Microsoft have many skilled engineers and don't need more people to fix problems" you might say . Well lets define a new concept "0 day vulnerability" it is a normal vulnerability ,nothing special, no supernatural power, but the issue with it is that it is well known to hackers as well as engineers but they didn't come out with a way to fix it , you see! even Microsoft suffers from 0day vulnerabilities especially in its project Internet Explorer that despite all the skill of its engineers those problems persist for a long time opening the door for hackers to access confidential informations all over the world. On the other side, Internet explorer's rivals like firefox or Chromium that are open source are pretty far from those issues .



### OK open source is secure, but what about OpenSSL security problem ?

The most influential flaw ever reported , with thousands of websites still vulnerable to it until this moment that I'm writing this article, is the famous heart Bleed vulnerability , well ... here is the answer of the programmer who claimed responsibility of this flaw in late 2011 "not the open source model is to blame" Robin Seggeleman said . Then why did the vulnerability persisted for so long? Because, in his view, OpenSSL, while widely deployed, is also under-funded. "OpenSSL is definitely under-resourced for its wide distribution. It has millions of users but only very few actually contribute to the project" he told the Guardian news ..

That is why many project managers have adopted the open source style, even for commercial purposes. Many companies went for “opening” their projects’ source codes to the public in order to benefit from the huge community that would contribute to it, and by the way open source is often said “ready” for such usage, because enterprises need easy deployment of its projects with highly scalable, agile and most importantly, secure deployment, those criterias are the base features of open source ..

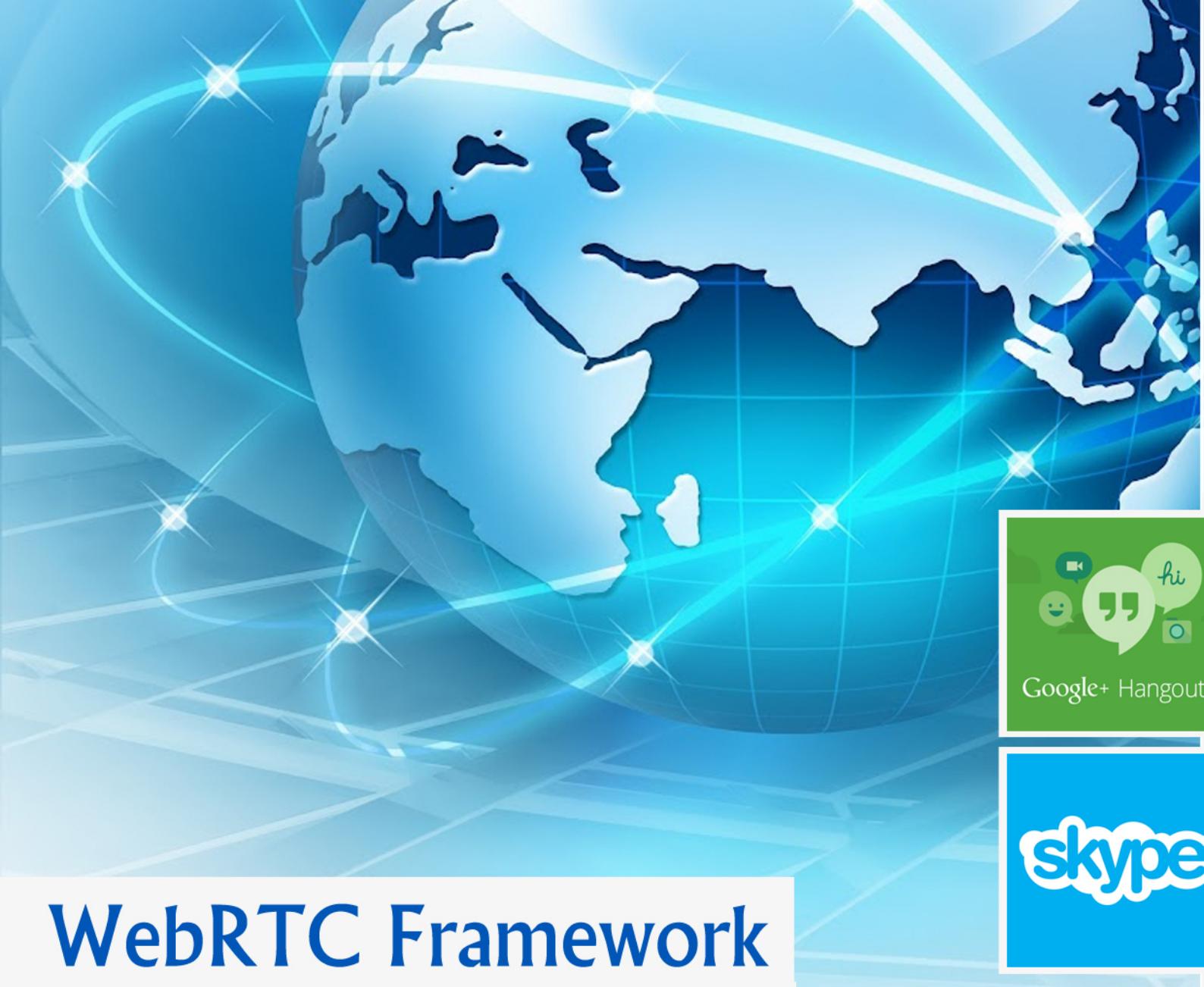
So next time you are stuck between open source or commercial solution you should not use security criteria as an excuse for choosing proprietary software ..

so lets sum up , both closed and open source software have their own concept of security, either by giving the source to the community to fix it or by preventing the community from hacking it .

Although closed source has a point in security, reality and facts make the open source the actual winner ..

Saif Farhani





# WebRTC Framework

Searching for a simple way to communicate with your friends ? Tired from installing plugins ?

Wondering about your privacy ? None of these questions meets your requirements ?

Imagine a world where your phone, TV and computer could all communicate on a common platform. Imagine it was easy to add video chat to your web application. That's the vision of WebRTC.

## So What's WebRTC ?

WebRTC stands for Web Real-Time Communication. It is a free, open project that enables audio/video streaming, text messaging and data sharing between browser clients. It is sponsored and maintained by big companies and communities such as Google and Mozilla. As a set of standards, WebRTC provides any modern browser like Firefox, Chrome and Opera with the ability to share application data and perform teleconferencing peer to peer. All these features are offered without the need to install any plug-ins for your browser and any third-party software in your operating system.





WebRTC is an application programming interface (API) that supports browser-to-browser applications for voice calling, video chat, and P2P file sharing without plugins. It is getting big steps in development during these last years. That doesn't mean that you have to wait in order to try the features. Currently in development and amelioration are the Network Stream API, which represents an audio or video data stream, and the PeerConnection API, which allows two or more users to communicate browser-to-browser.

That's not all! As said before, the purpose of WebRTC is to offer all the features a normal user wants, and a developer needs. So we can mention the under development DataChannel API that enables communication of other types of data for real-time gaming, text chat, file transfer, and so forth.

#### **Privacy, plugins and software setup :**

Skype, Hangouts and others exist and they offer a reliable service, so why to think for a replacement ?

The proprietary services such as Microsoft Skype or Google's Hangouts have a very well built-in tools for audio, video and text messaging, but they don't offer this without any price. In fact all the user's data is being stored at huge servers by these big companies for further access. They can use it for statistics, studies, or for any other concern. So the user

is omitted the right for privacy and is treated as a product.

Moreover, these services are more complex than WebRTC as the user has to necessarily install and/or configure a software or a plugin on his operating system in order to use them, where he just needs a modern web browser in order to benefit from WebRTC services as it is plugin free.

#### **What about privacy and security in WebRTC ?**

WebRTC has a native Security provided without any plugin. It offers mandatory encryption of all the exchanged data. In addition to that, designed for peer-to-peer communication and asking the user explicitly about access to Webcam and Microphone, WebRTC removes any fear for the user to have his connection intercepted by a hacker or have his personal and private data theft.

#### **So, are there any applications built using WebRTC ?**

The magic of WebRTC and its widespread adoption lies in the rich ecosystem of applications being built on it and the use-cases enabled by real-time communications on the web.

A full list of applications and services based on WebRTC is available here :

<http://www.webrtcworld.com/webrtc-list.aspx>

# Tutorial

After specifying the major characteristics of WebRTC it is time to give it a try. You can test a live demo simply by visiting <http://apprtc.webrtc.org/>, accepting to share your Camera and Microphone, and opening the generated link in a new tab in your browser or sending it to a friend. That's all what you need to do, you have successfully made it, you launched a conversation with audio and video between the two peers.

Depending on your needs and how deep you want to go down the WebRTC you can create your own server on your PC with few simple steps. First you should install Node.js to be able to run the server, the npm package manager, which will allow you to easily install modules and packages to use with Node.js, and git to perform Git operations and download the source code of WebRTC from github. To do that, you can launch a terminal and run the following commands : (for Ubuntu users)

```
sudo apt-get update  
sudo apt-get install nodejs npm git
```



Then you can download the sources of the demo and setup it by :

```
git clone git@github.com:  
webRTC/webrtc.io-demo.git  
cd webrtc.io-demo  
npm install  
cd site
```

and finally launch the NodeJS instance :

```
node server.js
```

In a version of Firefox or Chrome that has WebRTC support. Go to localhost:8080 and click allow to see your camera and enable the microphone Open a new tab or window, go to localhost:8080 and click allow to see your camera : the connection will be made between your two open tabs or windows.



## Want to start developing ?

This has been a little view on WebRTC and its impressive capabilities. If you are convinced enough to start developing, you can follow WebRTC conferences on [krangygeek.com](http://krangygeek.com) , find Google I/O videos concerning WebRTC in the Google Developers Channel on Youtube (<http://goo.gl/megBaQ>) to get started. You can also visit <https://github.com/webRTC/webRTC.io> and read the instructions.

## Conclusion :

Offering more features and flexibility than the well known audio and video communication services such as Skype and Hangouts, WebRTC combines security, efficiency and simplicity.

## Useful links:

<http://www.webrtc.org/>  
<https://github.com/webRTC>  
<http://docs.webplatform.org/wiki/apis/webrtc>

Oussama Krifa



# Raspberry Pi

## Gameboy Pocket (Pi-Pocket) by Travis Brown

The Raspberry Pi has taken the world by storm. The credit-card size computer has sold over two million units and has captured the imagination of kids and adults all over the globe and has allowed their creativity to soar. With a few peripherals and basic programming knowledge, your Raspberry Pi can do things you never imagined.

As we talked in the previous edition of “Ossec magazine” about what you need to start your own project now we can talk about what you can do with a RPi and how we can start building our first mini-project. This edition aims to help you get started as a coder by encouraging you to create your own projects. (Don't worry if you don't have a Raspberry Pi yet). So, we will start building a Game-boy Pocket (Pi-Pocket) and making the famous “Hello World!” program in RPi world. Why games? Because they're great way to learn programming: they're fun, you get to play them, and they include all ingredients common to most coding projects, from software utilities to robotic control. They can also lead to a great career as a game programmer.

### Description

This portable games console is capable of playing Game-boy games and other popular Linux (Doom, Duke Nukem..) using Raspberry Pi emulator. the Pi-Pocket provides over 3 hours of continuous play with unlimited play time while charging via the original power input.



RaspberryPi

### Case Modeling:

A Game-boy Pocket will be reused for this project. First, we will disassemble it and get sense for what can be useful and what else will make good additions.

### Game-pad:

The original Game-boy button pad will serve as the input to the RPi using a Teensy 2.0 that will provide simulated keyboard input over USB and other features. The Teensy USB Development Board is a complete USB-based microcontroller development system.

### Raspberry Pi:

We will try to free some space by removing the ports because the Pi with this thickness will not fit in the pocket. After removing we should boot our Pi to check if everything still working properly. Then we remove the linear regulator and the capacitor near the micro usb port. Finally we should ensure that the Pi fits inside the Game-boy.



Raspb



## Parts List

- \$35 - Raspberry Pi Model B (512MB RAM)
- \$25 - 2.5 Inch Composite LCD Screen
- \$20 - SanDisk Ultra Micro SDHC Class 10 30MB/s 32GB SD Card
- \$16 - Teensy 2.0 USB Development Board
- \$15 - Gameboy Pocket pretty much any condition with working button pads
- \$7 - 3.7V Li-Ion (Li-Poly) Flat Cell 606168 2600mAh Battery
- \$7 - 3.7V to 5V Step Up Regulator for Li-Ion Cells
- \$5 - 3.3V Switching Mode Step Down Power Regulator
- \$3 - Miniature Audio Amplifier
- \$2 - 5V Mini 1A Charging Board for 3.7V Li-Ion Cells
- \$2 - Mini round 28MM thin speaker
- \$2 - USB Port with No Flanges
- \$2 - Replacement Gameboy Pocket Screen Lens



## LCD Screen

We will use 2.5 inch screen because we didn't find 3 inch screens all of them require some sort of driver board which would be a project by itself.

## Power Supply

3 main power supplies will be used :

- 3.3V Switching regulator to decrease power consumption.
- 3.7V Li-Po battery to provide protection.
- A charging regulator to supply power to the battery from the 5v input.

## Sound

We found a difficult placing the original speaker in the pocket. So we will use a small amplifier with a 5V input and we will reuse the original Game-boy potentiometer.

## Putting It All Together

With some creative thinking , everything fits together perfectly so the Pi-Pocket will keep the same feel of the original Game-boy.

### Diagrams

The diagrams that describes how the sections are connected together are provided by the original creator of this project in his Site.

### Code

The code for the Teensy can be found on GitHub(<https://github.com/WarriorRocker/pi-pocket>).

SOURCE:

<http://www.xodustech.com/projects/raspberry-pi-gameboy-pocket>

## “Hello World!” of embedded systems :(Blinking LED)

One of the few things that separates the Pi from other SBC (Single Board Computer) is the ability to use the GPIO (General Purpose Input/Output) pins which can be set as HIGH or LOW to control any external device. All you need is a female or male jumper wire to get started. You can use a HDD IDE connector to get the job done.

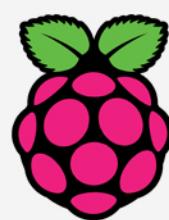
The fastest way to get started is to use python which comes pre-installed with all images. Download the RPi.GPIO library and copy the gz tar ball to the RPi wheezy Raspbian. Open the terminal and navigate to the extracted folder containing the RPi.GPIO library. Then type: \$ sudo python setup.py install to install the module.

### The Code :

```
import time # to delay the processor
import RPi.GPIO as GPIO # Raspbian build 4 comes with this module pre-installed

LED = 22 # assign a pin number to a variable
GPIO.setmode(GPIO.BOARD) # to use Raspberry Pi board pin number
GPIO.setup(11, GPIO.OUT) # set up GPIO output channel
GPIO.output(pin,GPIO.LOW) # the initial state

while True: #run forever
    GPIO.output(LED,GPIO.HIGH) #switch on the pin
    time.sleep(0.5) #wait
    GPIO.output(LED,GPIO.LOW) #switch off the pin
    time.sleep(0.5) #wait
```



RaspberryPi

Then, we save the file with the name “blinky.py” and open a new Terminal, next we move to the directory where we saved our file and we should see blinky.py if we type: \$ ls in command line. Finally we type: \$ sudo python blinky.py then enter and we should see our LED blinking.

Source: SparkFun Electronics (<http://www.youtube.com/watch?v=b6h95jNWg1g>)

# Fish

## friendly interactive shell

The fish shell is a modern, attractive, and powerful command shell that can extend the capabilities of the usual bash shell

### How to Install the Fish Shell

- sudo apt-get update
- sudo apt-get install fish

If your terminal has the ability to display colored output, you'll have noticed that your prompt is automatically colored as well. Furthermore, it is dynamically colored. If you type something that is not a valid command, it will show up as red.

### exp: this would be red:

- \$ ech

However, when you add the final "o", turning it into a command, you'll see it instantly turn green. This provides you with useful feedback that can make it easy to spot typos early. You may also notice that the TAB completion is excellent. So when you write cd and hit the TAB key fish intelligently lists only the directories for the cd command, since these are the only values that make sense.

finally, for full documentation write help in terminal then enter you will be redirected to the official documentation in your browser. So don't hesitate fish will save your time.

Atef Arfaoui

# Conky

## Set up your Conky !!

Conky is a lightweight software system monitor written in c language under GPL and BSD licenses. it makes you able to supervise system variables such as CPU's activity, memory status ... you can also display weather and news from official websites.

**-to install conky run in terminal :** sudo apt-get install conky-all

**-to run conky :** conky &

This is the default configuration located in /etc/conky/conky.conf

You can create .conkyrc file in your home directory and build your own script.

Or just copy the script conky.conf in the .conkyrc file and make your own configuration.

For example, if you want a transparent background, add those lines to .conkyrc :

own\_window\_argb\_visual true

own\_window\_transparent yes

**-to make conky run on startup (Gnome):**

**run :** gnome-session-properties

**add this script :**

```
#!/bin/sh
```

```
sleep 20s
```

```
conky
```

*There is a lot of cool, ready to use, conky scripts such as conky lua :*

<http://gnome-look.org/content/show.php/Conky+lua?content=139024>



# True-crypt

**Did hackers make your life miserable ? Are you looking for a safe environment to work in ?**

The open source community offers a simple but powerful tool to help you lock up your files starting from system drives to backup disks . It is the famous True-crypt , a free software that encrypts data “on-the-fly”. First of all , let's define the “on-the-fly” encryption : it's a specific kind of encryption that secures the information on a computer storage device , but it remains accessible to a verified user .In that way , the information is read and written while encoded and it's never stored in a device without protection . This method is very fast and completely automatic .In fact , the user has only to provide the authentication required to access the information . True-crypt is one of the most famous on-the-fly encryption applications that makes working with encrypted files as simple as working with regular ones. It is very simple to use and available for windows , vista , mac OS and Linux . Now let's have a quick look on how does true-crypt work . The first step , of course , is to download and install the software. And then you need to create a volume with an encrypted file container ,this is where the encrypted files are stored, you have also to set the size and the name of your container .

For more flexibility , you get to choose weather to keep it visible or hidden . Then you have to select your encryption algorithm, IE the way you want to encrypt your file as well as the speed of the encryption process . The most common ones are AES and RIPMD-160. Once you have configured your necessary space , you need to set a good password for your encrypted container . For more security you can choose a key file or combine both of them . Using only a password limits your security level since it's limited to the symbols on your key board. However the use of a key file increases the level of randomness . And that's how does True-crypt work . But don't get over excited , have you heard of the mysterious disappearance of this software ?

In Mai 2014 these warnings appeared on the official site of True crypt :

“WARNING: Using TrueCrypt is not secure as it may contain unfixed security issues”

“This page exists only to help migrate existing data encrypted by TrueCrypt.”

It has been said that Microsoft ended support for windows XP , as a result the anonymous developers of True-crypt shuttered the True-crypt site end warned users that it is no longer secure .

How true is that ? And have we lost this powerful tool forever ?

Hanin Slimene





"In real open source, you have the right to control your own destiny". Guess who said this , yes he is the principal force behind the development of the Linux kernel ..

A Finnish American software engineer called [Linus Torvalds](#) who was born on December 28th, 1969 ..

He later became the chief architect of [the Linux kernel](#) and now acts as the project's coordinator. He also created the revision control system Git as well as the diving log software Subsurface.

According to Wikipedia , he was honored, with the 2012 Millennium Technology Prize by the Technology Academy Finland "in recognition of his creation of a new open source operating system for computers leading to the widely used [Linux kernel](#)". He is also the recipient of the 2014 IEEE Computer Society Computer Pioneer Award.

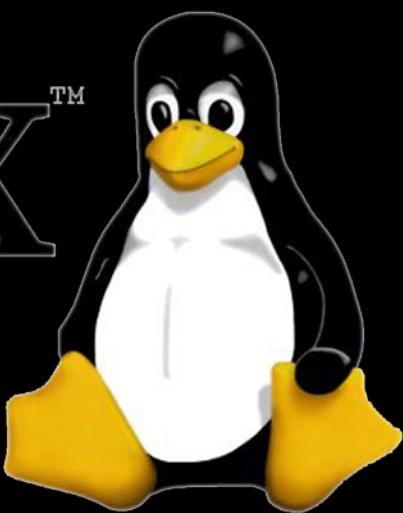
So if you say the word Linux , [Linus Trovalds](#) is the first person that should come to your mind ..

" The Linux philosophy is 'Laugh in the face of danger'. Oops. Wrong One. 'Do it yourself'. Yes, that's it."

Now you know a little bit of this man who made a huge impact , I hope this gives you a little push to look into it a little bit more , try to contribute and maybe one day you'll do something to be proud of like him ...

Lamia Ben Neticha

Linux™



# Quiz!

extrait : ossec scc 2014

**1** Vous voulez créer simplement dans votre répertoire personnel le chemin sources/C/ backup, sachant qu'aucun de ces répertoires n'existe :

- a- mkdir sources/C/backup
- b- mkdir -p sources/C/backup
- c- mkdir sources ; cd sources ; mkdir C ; cd C ; mkdir backup
- d- mkdir -p sources C backup

**2** En tant qu'utilisateur, pour lister l'intégralité des fichiers du système et placer tous les résultats quels qu'ils soient dans "liste", quelle doit être la bonne redirection ?

- a- 2>dev/null >liste
- b- >liste 2>liste
- c- 1>liste 2>&1
- d- 2>1 >liste

**3** Dans vi, laquelle des commandes suivantes supprime la ligne courante et les 16 lignes suivantes?

- a- 17d
- b- 17dd
- c- 17x
- d- d17d
- e- 16d

**4** En bash, l'insertion 1>&2 après une commande redirige :

- a- l'erreur standard vers l'entrée standard
- b- l'entrée standard vers l'erreur standard
- c- la sortie standard vers l'erreur standard
- d- la sortie standard vers l'entrée standard

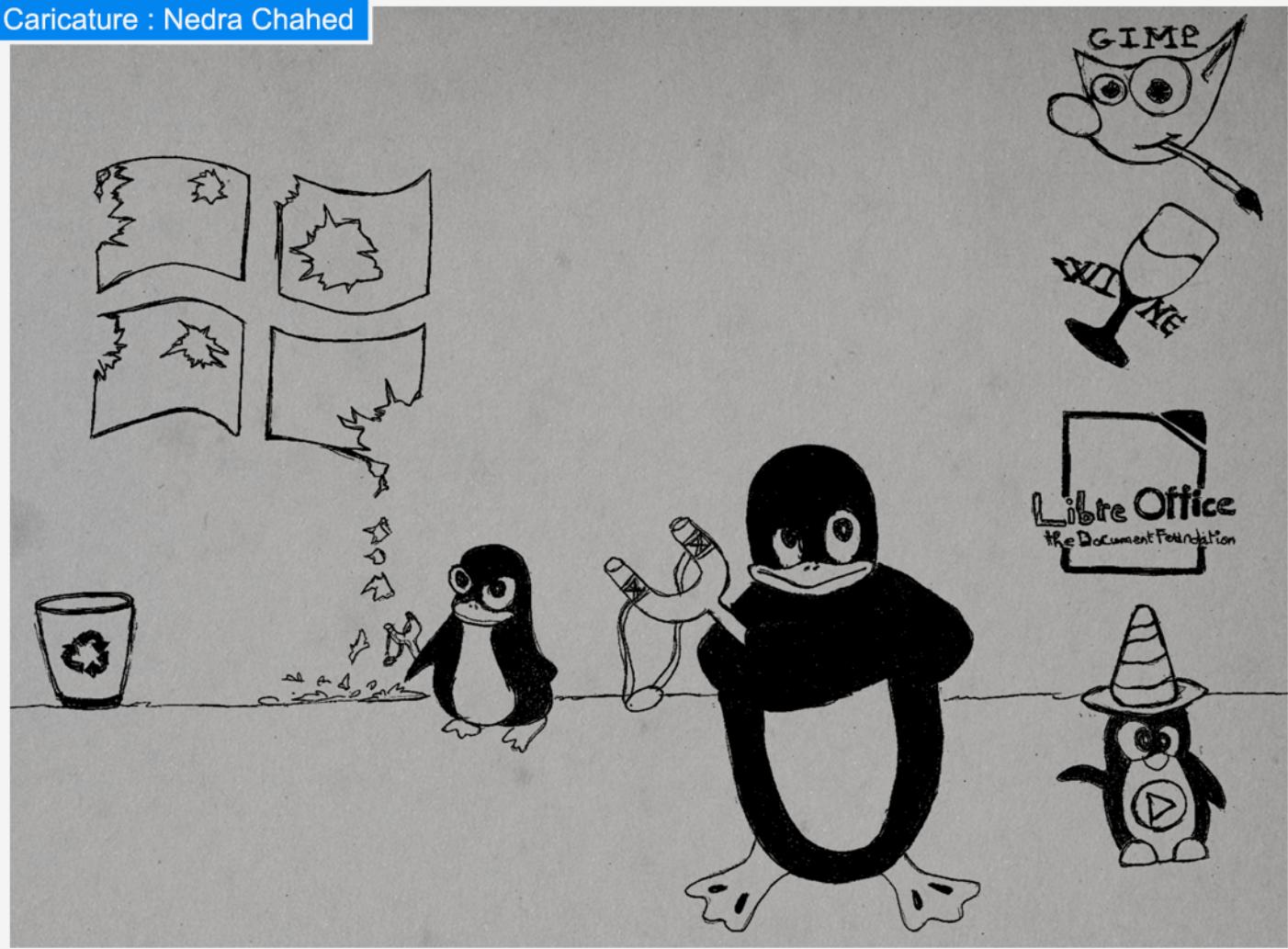
**5** Quelle ligne écrire dans un script pour sortir avec un message d'erreur et un code de retour 1 si le nombre de paramètres passés est égal à 0?

- a- test \$# != 0 || echo Erreur ; exit 1
- b- [ \$# -eq 0 ] && { echo Erreur ; exit 1 ; }
- c- test \$0 -ne 0 && ( echo Erreur; exit 1 )
- d- [[ \$# -eq 0 ]] && echo Erreur && exit 1

**6** Quelle commande va supprimer les lignes dupliquées d'un fichier trié?

- a- filtrer
- b- trim
- c- uniq
- d- wc

Caricature : Nedra Chahed



## Forum annuel

"Cloud computing et ses applications" sera le thème choisi pour la 9ème édition du Forum annuel des entreprises à l'Ecole Nationale des Sciences de l'Informatique (ENSI), organisé par l'ENSI Junior Entreprise, le 19 novembre 2014.

Rassemblant plus de 700 étudiants, 20 sociétés et plusieurs personnalités, ce rendez-vous présente une occasion incontournable où les étudiants tutoient leurs aînés, cadres et chefs d'entreprises qui s'activent dans le domaine des TIC.

Les présents pourront débattre sur le thème principal choisi, ou participer à des workshops portant sur différents aspects de l'informatique. Ceci dit, c'est aussi une occasion à la famille Junior de se rapprocher encore plus les uns des autres créant ainsi une atmosphère conviviale et hospitalière.



## Hackathon

You definitely wonder " what does the word Hackaton even mean ? " Well , your answer is here : it is an event in which computer programmers and people interested in software development come together to collaborate on the implementation of software projects concerning a given theme ..



" Ok , so now I know what is Hackaton, what does it have to do with me ? " The Open Source Software ENSI Club (OSSEC) is planning to organize such an event here on campus it's planned to last one week-end but I'm not a news spoiler so I can't tell you about the theme nor the date , sorry !..

"Annd , can I ask about the program ? "

Suure ! The event will begin with a series of conferences on Saturday afternoon then the beginning of the challenge between teams of three , the aim is to innovate and bring something new to the floor , the only condition is using open source technologies and of course the project will be open source ..

Come on, what are you waiting for ?! Start practising and gathering your team members for the big challenge .. The prize is worth it ;)

## About us :

OSSEC ( Open Source Software Ensi Club ) est un club créé à l'école nationale des sciences de l'informatique (ENSI) en 2011 ..

Notre but est de promouvoir la culture de l'open source entre les étudiants de l'ENSI , qui se base essentiellement sur le partage et l'échange .

Nous visons ainsi à assurer un environnement de travail et de développement qui prépare nos adhérents à la vie professionnelle ..

Plus de détails sur notre site : [www.ossec.tn](http://www.ossec.tn)



# --Team--

## Director :

— Lamia Ben Neticha

## Editors :

— Lamia Ben Neticha  
Ousama Krifa  
Hanin Slimene

## Designer :

— Haithem Abdelli

## Writers :

— Lamia Ben Neticha  
Ahmed Bessifi  
Chaima Guizani  
Ahmed Nefzaoui  
Ousama Krifa  
Atef Arfaoui  
Haithem Abdelli  
Saif Farhani  
Hanin Slimene  
Dhia Edine Chouchane

If you'd like to become part of the magazine's team  
please send us an e-mail :

[magazine@ossec.tn](mailto:magazine@ossec.tn)

Conventus



Mag  
N°03

ossec

