

No Arch Linux, menos é mais

# Do arco da velha

Para quem procura um sistema rápido, estável e moderno, sem pesadas interfaces gráficas para tudo, o Arch pode significar o fim dessa busca.

POR JON KENT

Ultimamente, as mais famosas distribuições têm dado ênfase ao uso do Linux como estação de trabalho, além da facilidade de instalar e configurar o sistema sem a necessidade de se aventurar por nenhuma linha de comando. Apesar do sucesso dessa iniciativa, um segmento significativo da comunidade Linux ainda prefere um ângulo mais “simples”. Esses usuários querem clareza, estabilidade e desempenho, sem dar a mínima para assistentes de configuração e ferramentas gráficas. Essa turma sempre gravitou em torno de distribuições como *Slackware*, *Gentoo* ou *Debian*. Mas um novo integrante dessa família vem ganhando cada vez mais peso: o *Arch Linux* [1].

A distribuição foi criada pelo canadense Judd Vinet em 2001, ao constatar que nenhum sistema satisfazia seus ideais. Assim nasceu o Arch, com muitos dos conceitos do Debian, Gentoo e Slackware. Gradualmente, a distribuição evoluiu e se tornou uma opção simples, poderosa e estável, com uma base fiel de usuários e desenvolvedores.

Como há poucas ferramentas de configuração, esse não é um sistema para usuários iniciantes. A filosofia do Arch é manter o usuário bem próximo da estrutura interna da distribuição. É necessário manipular diretamente arquivos de configuração, como nos “bons e velhos tempos”. Há vantagens nítidas sobre outras distribuições “simples”, como o Slackware. Por exem-



```

# Use COLOR: use ANSI color sequences in startup messages
#
# HARDWARELOCAL="localtime"
TIMEZONE=""
KEYMAP=""
CONSOLEFONT=""
CONSOLEMAP=""
RECCOLOR="yes"

# Scan for LVM volume groups at startup, required if you use LVM
#LVM="no"

#
# Networking
#
HOSTNAME="willy"

#
# Modules to load at boot-up (in this order)
# (prefix a module with a ! to disable it)
#
MODULES=(usbserial t4e-usb usbhid serial)

#
# Interfaces to start at boot-up (in this order)
# Declare each interface that first is INTERFACES
# (prefix an interface in INTERFACES with a ! to disable it)
#
# Note: to use DHCP, set your interface to be "dhcp" (setm="dhcp")
#
lo="lo 127.0.0.1"
eth0="eth 192.168.0.2 network 255.255.255.0 broadcast 192.168.0.255"
wlan0="dhcp"
INTERFACES=(lo eth0)

#
# Routes to start at boot-up (in this order)
# Declare each route that first is ROUTED
# (prefix a route to ROUTED with a ! to disable it)
#
gateway="default gw 192.168.0.1"
ROUTE=(gateway)

#
# Daemons to start at boot-up (in this order)
# (prefix a daemon with a ! to disable it)
# (prefix a daemon with a & to start it up in the background)
#
DAEMONS=(systemd logd netd ntpd networkd ntpd cron portmap rshd rsyncd
)

```

Figura 1: É preciso configurar todo o sistema editando os bons e velhos arquivos de configuração.

plo, o suporte *hotplug* e o gerenciamento mais eficiente de pacotes.

Além disso, o Arch é otimizado para processadores i686, trazendo mais desempenho que distribuições projetadas para rodar em uma gama maior de arquiteturas. Usuários do Arch garantem que o sistema possui “a estabilidade e a simplicidade do Slackware com a velocidade do Gentoo”. O **quadro 1** (“A comunidade explica”) contém mais comparativos com outras distribuições.

Embora seja difícil aprender a usar uma distribuição como o Arch, a vantagem é que uma vez que você domina a fera, adquire conhecimento profundo sobre Linux. Neste artigo vamos iniciá-lo no Arch, desde a instalação até o uso de seus recursos mais interessantes.

## Instalação

Ao contrário das grandes distribuições, o instalador do Arch é baseado em texto. Quem já instalou Slackware ou Debian se sentirá em casa. O Arch é instalado sem problemas ao lado de qualquer sistema operacional pré-existente – mas não deixe de fazer backups dos dados importantes por causa disso. Durante a instalação, vale a pena ter acesso a outro PC para consultar a boa documentação no site do Arch [3].

Como na maioria das instalações, primeiro é necessário particionar o disco. Isso pode ser feito com o *cfdisk*. Se preferir, é possível deixar o Arch tomar para si todo o disco rígido. Após criar as partições e definir os pontos de montagem, é preciso selecionar os pacotes.

O recomendável é que você instale os pacotes básicos nesse estágio, deixando o restante para quando o sistema já estiver rodando corretamente.

Depois que o sistema básico estiver instalado, escolha um kernel configurado para *IDE* ou *SCSI* (que você vai precisar, caso tenha dispositivos *SATA*). É possível recompilar um kernel mais a seu gosto, se preferir. Contudo, nesse estágio, faz mais sentido escolher um kernel pronto antes de fazer uma mudança drástica como essa. Opte por um kernel *udev*, mais atual, em vez do *devfs* – o Arch já trabalha com o novo *udev*. O último estágio é a configuração do sistema. É nesse ponto que o acesso à documentação oficial vem a calhar. ➡

### Quadro 1: A comunidade explica

Uma seção no site oficial do Arch compara a distribuição com outros sistemas populares [2]. Obviamente, a análise foi feita pelos desenvolvedores do Arch – os de outras distribuições devem ver a coisa de maneira diferente. Mas o texto fornece uma boa idéia dos objetivos do Arch.

#### Gentoo

O Gentoo tem mais programas disponíveis para instalação. Já o Arch permite a distribuição tanto de binários quanto do código fonte. Os pacotes baseados em código fonte no Arch são mais fáceis de serem criados do que os *ebuilds*. O Gentoo é mais “portável” em uma instalação padrão, já que os pacotes são compilados para sua arquitetura específica. O Arch traz pacotes compilados apenas para a arquitetura i686 (embora versões para i586 e x64 estejam a caminho). Não há nenhuma prova documental de que o Gentoo seja mais rápido que o Arch.

#### Crux

O Arch descende do *Crux*. Judd Vinet certa vez resumiu as diferenças entre eles: “Eu usava *Crux* antes de criar o Arch. O Arch começou, na verdade, como um *Crux*. Então criei o *pacman* e o *makepkg* para substituir meus pseudo-scripts em bash (comecei o Arch como um sistema *Linux From Scratch*) para gerenciamento de pacotes. As duas são distribuições completamente separadas mas, tecnicamente, são quase a mesma coisa. Um exemplo: ambas, oficialmente, têm meios de resolver dependências – embora o *Crux* tenha uma comunidade que fornece outros recursos. O *prt-get* do *Crux* chega a dar conta de uma lógica de dependências, embora rudimentar. O *Crux* tende a ignorar muitos dos problemas que também temos, já que é um conjunto de pacotes bem minimalista”.

#### Slackware

O Slackware e o Arch são ambas distribuições “simples”. As duas usam scripts de inicialização do tipo *BSD*. Mas o Arch tem um sistema de geren-

ciamento de pacotes muito mais robusto, o *pacman*. Ao contrário do que é possível fazer com as ferramentas-padrão do Slackware, o *pacman* permite a atualização do sistema todo de forma simples. O Slackware tem um ciclo de lançamentos mais conservador, preferindo incluir programas em versões comprovadamente estáveis. Nesse aspecto, o Arch é muito mais “atual”. O Arch roda apenas na arquitetura i686, enquanto o Slack pode rodar em sistemas i486. Resumindo, o Arch é um sistema muito bom para usuários do Slack que querem um gerenciamento de pacotes mais robusto e programas mais atuais.

#### Debian

O Arch é mais simples que o Debian. Há menos redundância de programas e melhor estrutura para a construção de pacotes personalizados. O Arch é também mais permissivo no que se refere a pacotes “não-livres”, conforme a definição *GNU*. O Arch é otimizado para i686. E os programas são mais atuais que os do Debian.

#### Distribuições “gráficas”

As distribuições gráficas são bem parecidas, e o Arch é bem diferente delas. Ele é baseado em texto e na linha de comando. Para quem quer aprender Linux, o Arch vence. As distribuições gráficas vêm com interfaces de instalação (como o *Anaconda* do *Fedora*) e interfaces de configuração (como o *YaST* do *SuSE*).

#### Arch X Ubuntu

O Arch tem uma base estrutural mais simples que a do *Ubuntu*. Se você quiser compilar seu próprio kernel, testar projetos CVS, ou compilar outros programas de vez em quando, o Arch é melhor. Se quiser um sistema pronto e produtivo bem rápido, sem precisar mexer nas tripas do sistema, o *Ubuntu* sai ganhando. No geral, desenvolvedores e fuçadores vão preferir o Arch ao *Ubuntu*.

Será apresentada uma lista de arquivos de configuração que precisam ser editados. Há comentários úteis dentro desses arquivos, mas ajuda muito se a pessoa souber a função de cada um deles. Quem já tiver configurado sistemas Linux, sem a ajuda do instalador gráfico, não vai estranhar essa parte. Por exemplo, o `rc.conf` contém as configurações de rede, nome da máquina, módulos do kernel e serviços a serem iniciados. Editores como o *vi* ou *nano* são fundamentais nessa etapa.

A ordem dos serviços no `rc.conf` é importante, já que eles são iniciados exatamente na ordem em que foram colocados. Não há checagem de interdependência de serviços. Então, se um deles não iniciar, vale a pena verificar se a ordem está correta. Por último, verifique (e depois verifique novamente) as mudanças que você fez. O instalador vai prosseguir mesmo que você não tenha editado todos os arquivos importantes – nesse caso, certamente ocorrerão erros.

Quando terminar o processo de instalação, reinicie a máquina. Agora já temos um sistema Arch mínimo. O próximo passo é atualizar os pacotes com a ferramenta *pacman* (mais informações sobre isso a seguir), antes de começar a instalar novos programas.

Se você escolheu o *udev* no lugar do *devfs*, não haverá problema nenhuma com esse tipo de atualização. Já, se escolheu *devfs*, serão necessárias etapas adicionais para converter o sistema *devfs* para *udev*.



**Figura 2:** Embora o foco do Arch seja a configuração manual, depois que isso é feito, o sistema torna-se tão confortável quanto qualquer outra distribuição com *Gnome* ou *KDE*.

**Tabela 1: Scripts de inicialização do Arch**

Arquivo	Função
<code>/etc/rc.sysinit</code>	Responsável pelo carregamento e configuração do sistema
<code>/etc/rc.single</code>	Script para o nível de sistema de usuário único (single user)
<code>/etc/rc.multi</code>	Nível multiusuário
<code>/etc/rc.local</code>	Nível multiusuário local
<code>/etc/rc.shutdown</code>	Script para desligar o sistema
<code>/etc/rc.d/*</code>	Daemons configurados no sistema

## Inicialização

A principal filosofia do Arch é dar ao usuário o controle completo da configuração do sistema. Como vimos durante a instalação, nada é configurado automaticamente e nenhum serviço é acionado, a não ser que você mesmo o faça. Isso permite que os usuários entendam o funcionamento do Arch, e do próprio Linux, bem rápido.

O Arch usa o sistema de inicialização do tipo BSD, também usado no Slackware. Para alguns, esse é “O Verdadeiro Sistema de Inicialização”, embora outros prefiram o *System V*, usado por um número maior de distribuições Linux. Esse último sistema também pode ser usado, e seus scripts estão em `/etc/rc.d`.

## Gerenciamento de pacotes

A ferramenta para gerenciar pacotes binários é o *pacman*, equivalente ao *apt-get* do Debian. O formato dos pacotes é *tar.gz* e o banco de dados é baseado em texto.

Como no *apt-get*, o *pacman* permite a instalação/remoção de pacotes, consultas ao status de determinados programas, atualização do banco de dados... O *pacman* também permite a utilização tanto do repositório oficial quanto os de usuários.

Os repositórios de usuários, ou AUR (*Arch User Repositories*) [4] são um recurso bem útil, permitindo que um usuário disponibilize um pacote ausente do repositório padrão. Um bom exemplo disso é o *fouiny\_repo*, que contém pacotes da versão E17 do *Enlightenment*. É possível criar seu próprio repositório com os pacotes que você queira compartilhar com a comunidade.

Como qualquer outro bom gerenciador de pacotes, o *pacman* permite que você faça um upgrade geral do sistema de maneira fácil. Basta sincronizar a lista de pacotes de seu sistema com a do repositório. Então, todos os pacotes serão atualizados para suas últimas versões. Instalar também é fácil: todas as dependências serão resolvidas e baixadas automaticamente.



**Listagem 1: /etc/pacman.conf**

```
[repository-name]
Server = ftp://server.net/repo

[current]
# Add your preferred servers here, they will be used first
Include = /etc/pacman.d/current
```

O pacman é configurado no arquivo `/etc/pacman.conf`. Nesse arquivo, é possível especificar os repositórios. Ele também pode especificar que arquivos de configuração não devem ser modificados por nenhuma instalação, além de poder “congelar” a versão de um pacote instalado, para que ele não seja atualizado nunca mais. Por exemplo:

```
NoUpgrade = etc/passwd etc/group etc/shadow etc/sudoers
HoldPkg = pacman glibc
```

Na seção de repositórios, é possível defini-los tanto diretamente quanto “chamar” outro arquivo. Essa última opção é útil para os repositórios oficiais, que contam com muitos espelhos, conforme exemplifica a **listagem 3**.



**Figura 3:** A interface gráfica Enlightenment E17 está disponível para download no `fouiny_repo`, um dos repositórios de usuários do Arch, ou AUR (Arch User Repositories).

**Código fonte**

O Arch também fornece uma ferramenta para gerenciamento de pacotes com código fonte. Trata-se do ABS (Arch Build System), que é quase igual ao `emerge` do Gentoo. Ele foi projetado para empacotar programas fresquinhos (que ainda não es-



# GO-Global

Suas aplicações na Web em minutos.

**Web Enabling**

O GO-Global® publica na Web suas aplicações, legadas ou não (Windows®, Unix® e Linux®), sem a necessidade de reescrever uma linha de código sequer, mantendo todas as funções originais e preservando os investimentos anteriores em software e hardware.

**Virtual Office**

O GO-GLOBAL® é 100% seguro! Agora sim seus colaboradores poderão acessar seus aplicativos corporativos de qualquer lugar do planeta, a partir de qualquer computador, como se estivessem conectados localmente.

**Server Based Computing**

Com o protocolo exclusivo Rapid X, o GO-GLOBAL® dispensa a aquisição de licenças adicionais (Terminal Server) exigidas pelas demais soluções Server Based Computing.

A solução Rápida, Simples e Econômica.



Faça o download gratuito  
por 30 dias  
[www.go-global.com.br](http://www.go-global.com.br)



**ARC**  
SYSTEM  
Telefone: (11) 4153-5850  
[comercial@arcsystem.com.br](mailto:comercial@arcsystem.com.br)

Tabela 2: Opções do pacman

Opção	Função
-Sy	Sincroniza a lista de pacotes
-S pacote	Instala, reinstala ou atualiza um pacote
-S extra/pacote	Instala um pacote do repositório extra
-Su	Atualiza todos os pacotes instalados
-A /<caminho>/ pacote-version.pkg.tar.gz	Instala um pacote local
-R pacote	Remove um pacote
-Rs pacote	Remove um pacote e suas dependências, se possível
-Ss pacote	Procura por um pacote
-Si pacote	Exibe as informações de um pacote
-Scc	Limpa o cache dos pacotes baixados

tão disponíveis em forma binária), para customizar pacotes existentes de acordo com seus parâmetros ou até recompilar todo o sistema, usando suas próprias *flags* de compilação.

O ABS constrói pacotes binários que podem ser instalados pelo pacman. O uso do ABS não é obrigatório, mas ele permite que você faça ajustes finos na compilação de programas.

O ABS depende do *cvsup* e do *wget*. Portanto, eles precisam estar instalados, antes de você começar a usá-lo em compilações. Para instalar esses pacotes, simplesmente digite:

```
pacman -Sy cvsup wget
```

Vale a pena sempre usar a opção *-Sy* para se instalar programas. Isso garante a instalação das últimas versões, já que a opção atualiza a lista de pacotes antes da instalação.

Use o comando *abs* para sincronizar sua árvore ABS com a do servidor CVS do Arch, que fica espelhada localmente, em */var/abs*. A estrutura do diretório é bem simples. Em */var/abs* cada diretório se refere a um programa, contendo um arquivo *PKGBUILD* para a compilação do binário.

Para instalar programas com o ABS, entre no diretório do pacote desejado e execute o comando *makepkg* (isso exige que haja um arquivo *PKGBUILD* no diretório local). Após a compilação do código fonte, é possível instalá-lo com o pacman:

```
Instalar novo pacote:  
pacman -A pacote.pkg.tar.gz
```

```
Atualizar pacote:  
pacman -U pacote.pkg.tar.gz
```

Além desses diretórios, na raiz de */var/abs* há um diretório chamado *local*. Ele serve para a compilação com suas próprias *flags*. A idéia é que talvez esses arquivos prontos do tipo *PKGBUILD* não contenham as configurações que você precisa. Para usar esse recurso, crie um diretório em *local*, copie o arquivo para lá, acrescente ou remova configurações e rode o *makepkg* para compilar e instalar o programa.

Para controlar quais otimizações do *gcc* você quer, o *makepkg* usa um arquivo de configuração, o */etc/makepkg.conf*. Se você já usou Gentoo, as opções dentro desse arquivo não serão estranhas. Por padrão, o Arch usa a opção *-march=i686*

*-O2 -pipe*. Se você gosta de viver perigosamente, experimente a opção *-O3*. Como o Arch é firmemente baseado em *i686*, faz pouco sentido mudar a opção *-march*, ao menos que você esteja com delírios de aventura.

## hwd

Por padrão, o Arch usa os scripts de detecção de hardware do tipo *hotplug*, os mesmos da maioria das distribuições. Esse sistema simplifica os módulos de configuração, detecta dispositivos automaticamente e carrega os módulos necessários. Contudo, os desenvolvedores do Arch consideram o *hotplug* muito lento. Então, foi criado um caminho próprio: chamado *hwd*. Ao contrário do *hotplug*, o *hwd* executa o *modprobe* em processos-filhos, para que ele não tenha que esperar o *modprobe* carregar cada módulo, antes de continuar.

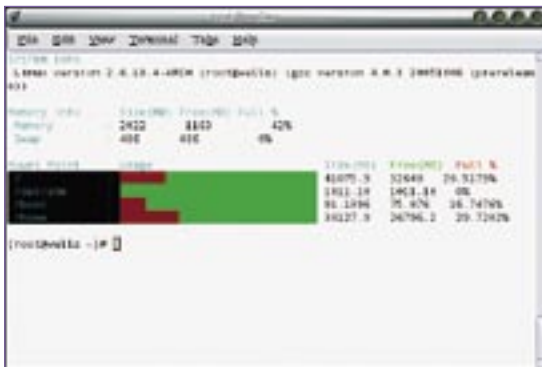
O *hwd* funciona tanto com o *devfs* quanto com o *udev*. Como o *hwd* não é uma ferramenta de configuração, ele não altera arquivos no */etc*. Isso iria contrariar a filosofia do Arch. Ao invés disso, ele detecta os componentes e fornece informações sobre como configurar manualmente os arquivos.

Caso você esteja satisfeito com o *hotplug*, não é necessário instalar o *hwd*. Essa também é outra filosofia do Arch, um componente extra é opcional de verdade. Se o computador raramente é reiniciado, ou se você consegue viver com o tempo de boot um pouco maior do *hotplug*, não há motivo para descartá-lo e partir para o *hwd*.

Mas se quiser experimentar, digite:

```
pacman -Sy hwd lshwd
```

Agora é necessário configurar o serviço *hwd* para ser iniciado junto com o sistema e desabilitar o *hotplug*. Como você já deve imaginar, o arquivo a ser



**Figura 4:** O hwd detecta componentes de hardware e mostra informações sobre o sistema.

editado é o `rc.conf`. Basta acrescentar a seguinte linha:

```
!hotplug hwd
```

O ponto de exclamação (“!”) na frente de `hotplug` desabilita esse serviço. O próximo passo é baixar as últimas tabelas `pci/pmcia`, que o `hwd` usa para identificar o hardware. Faça isso com:

```
hwd -u
```

Reinicie a máquina. No lugar do `hotplug`, agora quem vai subir é o `hwd`.

Além da detecção de hardware, o `hwd` e o `lshwd` podem ser usados também para preparar um arquivo de configuração inicial do `X`, ou ajudá-lo a configurar o `X`:

```
hwd -x
```

Esse comando vai criar um arquivo de exemplo do `X.org` em `/etc/X11`, que pode ser usado como base para o arquivo final.

## Ambiente gráfico

O Arch usa o `X.org` como servidor `X` e possui pacotes para todos os grandes ambientes gráficos para o usuário: `KDE`, `Gnome` e `XFCE`. Para cada um, há todos os aplicativos mais popula-

res, incluindo programas proprietários como o plugin do *Macromedia Flash* para o *Mozilla* e o leitor de PDFs *Acrobat Reader*.

Alguns programas vão entrar automaticamente no menu principal do ambiente escolhido, outros exigem que isso seja feito manualmente. Nesse quesito ainda não há padrões para o Arch, o que irrita um pouco. No

entanto, a velocidade com que o `Gnome` ou o `E17` rodam é impressionante – consequência da otimização para `i686` de toda a base funcional do Arch.

## Conclusão

O Arch é uma distribuição rápida e leve. E não há nenhum componente que contrarie o objetivo do projeto. Nesse sentido, é uma das melhores distribuições. Embora não haja ferramentas de configuração, essa política obriga o usuário a ganhar um entendimento profundo sobre o Linux. Se você quer controle absoluto sobre sua instalação, o Arch é uma ótima opção.

Embora não se trate de uma distribuição para iniciantes, a boa documentação e os fóruns amigáveis [5] permitem que você ponha a mão na massa sem muito receio. Mesmo para quem tem pouca experiência com Linux, essa é uma opção que vale a pena. ■

## INFORMAÇÕES

[1] Arch Linux: [www.archlinux.org](http://www.archlinux.org)

[2] Arch X Outros: [wiki.archlinux.org/index.php/Arch\\_vs\\_Others](http://wiki.archlinux.org/index.php/Arch_vs_Others)

[3] Guia de instalação: [archlinux.org/docs/en/guide/install/arch-install-guide.html](http://archlinux.org/docs/en/guide/install/arch-install-guide.html)

[4] Repositórios de usuários: [user-contributions.org/home/index.php](http://user-contributions.org/home/index.php)

[5] Em português: [tinyurl.com/cqy9f](http://tinyurl.com/cqy9f)

## A 4Linux apresenta

Cursos para embarcar em uma carreira de sucesso



Sua rede está realmente segura?



conheça: "Segurança em Servidores Linux usando a BS7799"

Você sabe medir as vulnerabilidades e ameaças de sua rede?



conheça: "Pen-Test - Análise e Testes de Vulnerabilidades em Redes Corporativas"

Seu site tem conteúdo dinâmico e interativo?



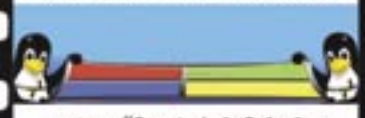
conheça: "Desenvolvendo Aplicativos WEB com PHP e MySQL"

Seu servidor de e-mail está seguro?



conheça: "Gmail seguro para Corporações"

Quer expandir sua rede Windows?



conheça: "Construindo Soluções Samba com Implementações Reais"

...aguarde outros cursos:

"Infra-estrutura Web com LAMP"

"Groupware com Software Livre"

"Perícia Forense usando Software Livre"

**4 LINUX**  
YOUR INTELLIGENCE IN LINUX  
FONE: 11. 2125-4747