FREE DVD fedora 23 Workstation 64-bit & ubuntu 15.10 Desktop Edition *"Wily Werewolf"* 32-bit

SCRIPTING TRICKS
System monitoring with custom Bash scripts

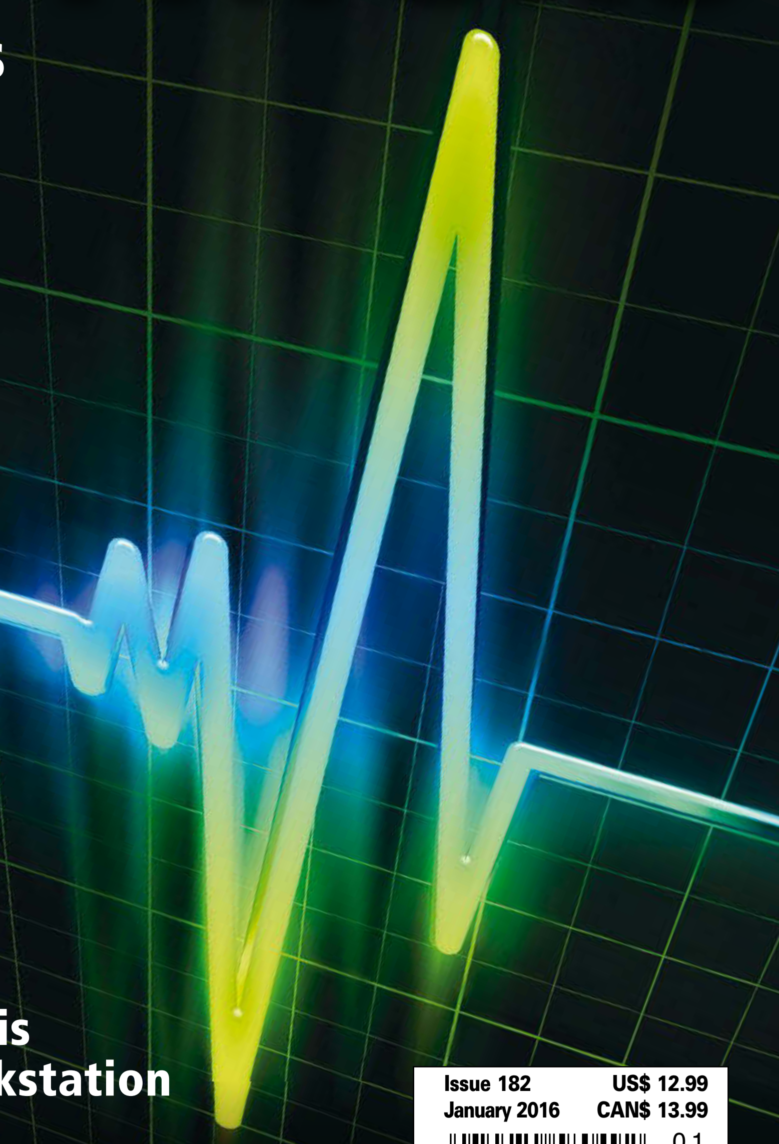
LINUX PRO MAGAZINE

JUST RELEASED!

JANUARY 2016

SCRIPT TRICKS

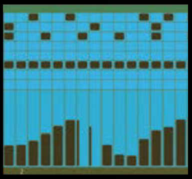
Create custom Bash scripts for system monitoring



Canvas Fingerprinting
HTML5 unleashes a weird new ad-tracking technique

Hash Functions
Are your stored passwords really safe?

Lynis
Get a convenient report of security problems



Tracktion T6
Make some music with this cross-platform audio workstation

Kubernetes
Wrangle containers in Red Hat's Project Atomic

Systemd
Essential commands for the new init

Julia
Cool new coding tongue for high performance

Issue 182 January 2016
US$ 12.99 CAN$ 13.99
0 74470 58049 2 01

WWW.LINUXPROMAGAZINE.COM

• Top performance
• Triple-redundant storage
• Super speedy SSDs
• guaranteed RAM
• Snapshot function
• Free Setup

DISTROS AND DVDs

Dear Linux Pro Reader,

We don't really know what sells a magazine on the newsstand. Big publishing companies have whole teams of marketing analysts studying trends and performing Big Data maneuvers on the sales data. Our ragtag group can only treat it as something of an art form: We throw an issue together, and sometime later, we find out if we guessed right when choosing what to put in it.

When I say sometime later, I mean a long time later. In the UK, it takes around 6 months to get final data on sales, and it can take up to a year to find out how we did in the USA and rest of world. Not that sales data is really so interesting as an editorial topic. The main reason I look at old sales records is to compare our magazine against itself – to see which issues sold best and try to understand why. Every time you reach for a magazine at the newsstand and buy it, you are voting for something in it. If an issue sells well, we ask, "What did they like about it?" The articles? The cover image? The cover headline? Or was it the illusive DVD?

We've never really had a scientific theory for choosing which distro DVD to include with each issue, although I have to admit, it used to be easier than it is now. A walk to our bookshelf sends me spinning down memory lane to my first two years on the job. When I started working for the magazine in 2004, the Linux world was just starting to recover from the shock of losing the popular Red Hat Linux free distro in 2003 when Red Hat created the Red Hat Enterprise subscription product and embraced Fedora as a free alternative.

RHEL was never really friendly with the idea of mass distribution through magazines, but Fedora joined the rotation with Fedora Core 3 in Issue 50. SUSE had not yet executed its own bifurcation into SLE and openSUSE, SUSE Linux was a big seller for us in those days. One of the articles in Issue 52 was a portentous little piece on a new startup distro that sported its own "philosophy" based on African conceptions of unity and oneness. Ubuntu had arrived, and it quickly gained momentum. We were there with an Ubuntu 5.04 "Hoary Hedgehog" DVD in Issue 53, and Ubuntu stayed around because it was (and is) something readers still seem to want. Another popular perennial from that era that is still in the lineup is Knoppix, the packed and powerful Live distro loved by hobbyists and admins alike. Other distros from past DVDs have disappeared or faded from the limelight – Yoper, Mepis, Linspire. We could have told the Xandros people their distro was in for a rocky ride just based on the comparative sales data for Issue 56.

We've always loved exposing readers to interesting new initiatives, and we have included some winners in our DVD series early in their history, such as Ubuntu, CentOS, and Linux Mint, but some of the other DVDs didn't do so well – and we stopped including them because you, our readers, didn't vote for them.

Of course, many interesting Linux distributions are designed for a special purpose and were never intended for a mass market audience – audio production distributions, for instance, or penetration testing systems. One important development that has allowed us to roll more of these specialized systems into the mix is the falling price for double-sided DVD-10 discs, which makes it easier to bind in one of the mainstream meat-and-potatoes Linux systems and still let our readers experiment with the lesser-known alternatives. But then, many of our readers still use 32-bit systems, and others definitely opt for 64-bit distros, so our DVD mix must accommodate both factions.

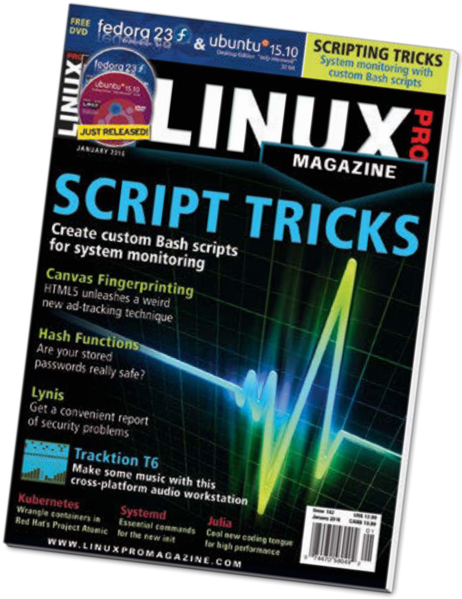
We're pretty sure you still want us to send new releases of Ubuntu, Fedora, and Knoppix. A major Debian only happens once in a while, and we think you'll want that one too. Mint tops the list at Distrowatch (by a lot), and we think it tops the list for many of you as well. What else? openSUSE? Mageia? Arch? Sabayon? What's your favorite Ubuntu flavor? Are you into troubleshooting? Gaming? Multimedia?

You can help us figure out what to put on our DVDs by letting us know what you want. Drop us a line at edit@linux-magazine.com and give us your suggestions for upcoming DVDs. Be sure to let us know if you're using a 32-bit or 64-bit system.

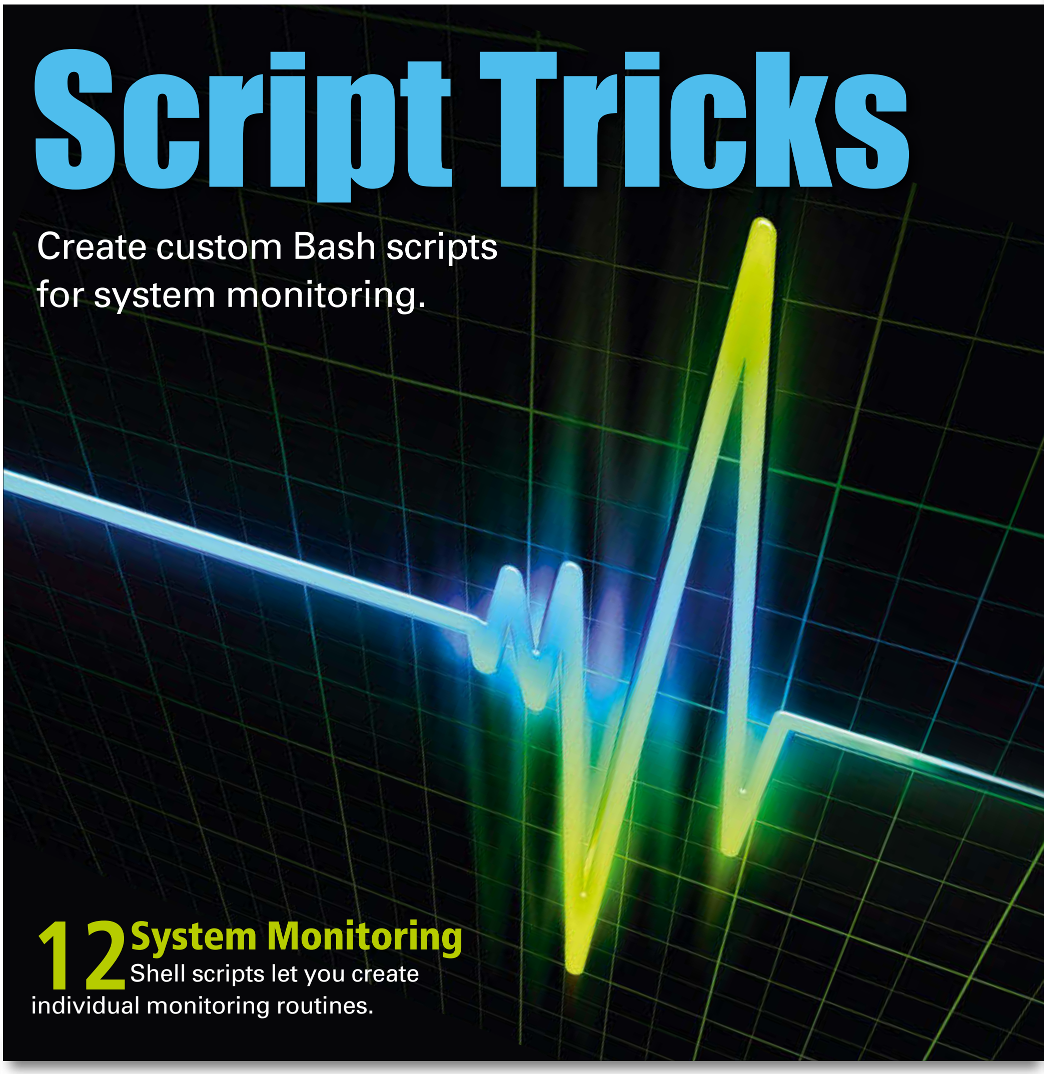
Joe

Joe Casad,
Editor in Chief

LINUX PRO MAGAZINE


FREE DVD fedora 23 & ubuntu 15.10
SCRIPTING TRICKS
System monitoring with custom Bash scripts
LINUX PRO MAGAZINE
SCRIPT TRICKS
Create custom Bash scripts for system monitoring
Canvas Fingerprinting
HTML5 unleashes a weird new ad-tracking technique
Hash Functions
Are your stored passwords really safe?
Lynis
Get a convenient report of security problems
Tracktion T6
Make some music with this cross-platform audio workstation
Kubernetes
Systemd
Julia
WWW.LINUXPROMAGAZINE.COM

news



Script Tricks

Create custom Bash scripts for system monitoring.

SERVICE

LINUX PRO MAGAZINE (ISSN 1752-9050) is published monthly by Linux New Media USA, LLC, 616 Kentucky St., Lawrence, KS, 66044, USA. Periodicals Postage paid at Lawrence, KS and additional mailing offices. Ride-Along Enclosed. POSTMASTER: Please send address changes to Linux Pro Magazine, 616 Kentucky St., Lawrence, KS 66044, USA.

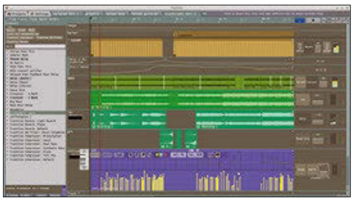
Community Notebook

HIGHLIGHTS

FEATURES

REVIEW

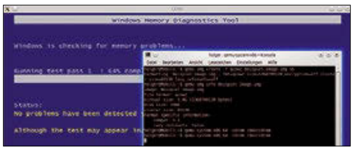
LINUXUSER



Little_Mejiro_Book.html - /home/tmp-pg/LBOOKS/Little_Mejiro_Book - Atom

- Little_Mejiro_Book
  - _assets
  - fonts
  - images
  - pages
  - .gitignore
  - bookcover.jpg
  - metadata.yaml
  - stylesheet.css
  - tasks.todo

- Little Mejiro Book
- ~~About This Book~~
- About Mejiro
- Install Mejiro
  - Installation on the Apache Server
  - Installation on the lighttpd Server
  - Install Mejiro on Raspberry Pi



```
Windows Memory Diagnostics Tool

Windows is checking for memory problems...

Status:
No problems have been detected

Although the test may appear
```

On the DVD

Fedora 23 Workstation (64-bit)

This month's 64-bit offering is Fedora Workstation [1] based on Gnome 3. Fedora 23 comes with all the languages, tools, utilities, and virtualization solutions you need to get going and be productive out of the box. Many packages are now hardened against potential security vulnerabilities [2]. Updates include Perl 5.22 and Python 3, as well as new versions of other popular desktop and developer applications. The SSL 3.0 protocol and RC4 cipher are disabled by default in components that use the system-wide cryptography policy (i.e., gnutls, openssl libraries, and all applications based on them).

Ubuntu 15.10 Desktop (32-bit)

The "Wily Werewolf" sports Linux kernel 4.2 and sees significant changes with the move into the systemd init system [3]. Various software has been updated, including Firefox 41, Gnome 3.16, MATE 1.10, and LibreOffice 5.0.2. A preview of the new converged phone, desktop, and tablet interface is available by logging in to a Unity8 session, where you can test the new windowed mode, "which allows users to multitask between multiple running apps" [4]. Whether everyday user or high-end developer, Ubuntu 15.10 has the modern tools you need.

TWO TERRIFIC DISTROS

DOUBLE-SIDED DVD!

ADDITIONAL RESOURCES

[1] Fedora Workstation: *https://getfedora.org/en/workstation/*

[2] Harden all packages: *https://fedoraproject.org/wiki/Changes/Harden_All_Packages*

[3] What's New in Ubuntu 15.10: *https://insights.ubuntu.com/2015/10/22/whats-new-in-ubuntu-15-10-desktop-and-devices/*

[4] Ubuntu 15.10 release notes: *https://wiki.ubuntu.com/WilyWerewolf/ReleaseNotes#New_features_in_15.10*

Defective discs will be replaced. Please send an email to cs@linuxpromagazine.com.

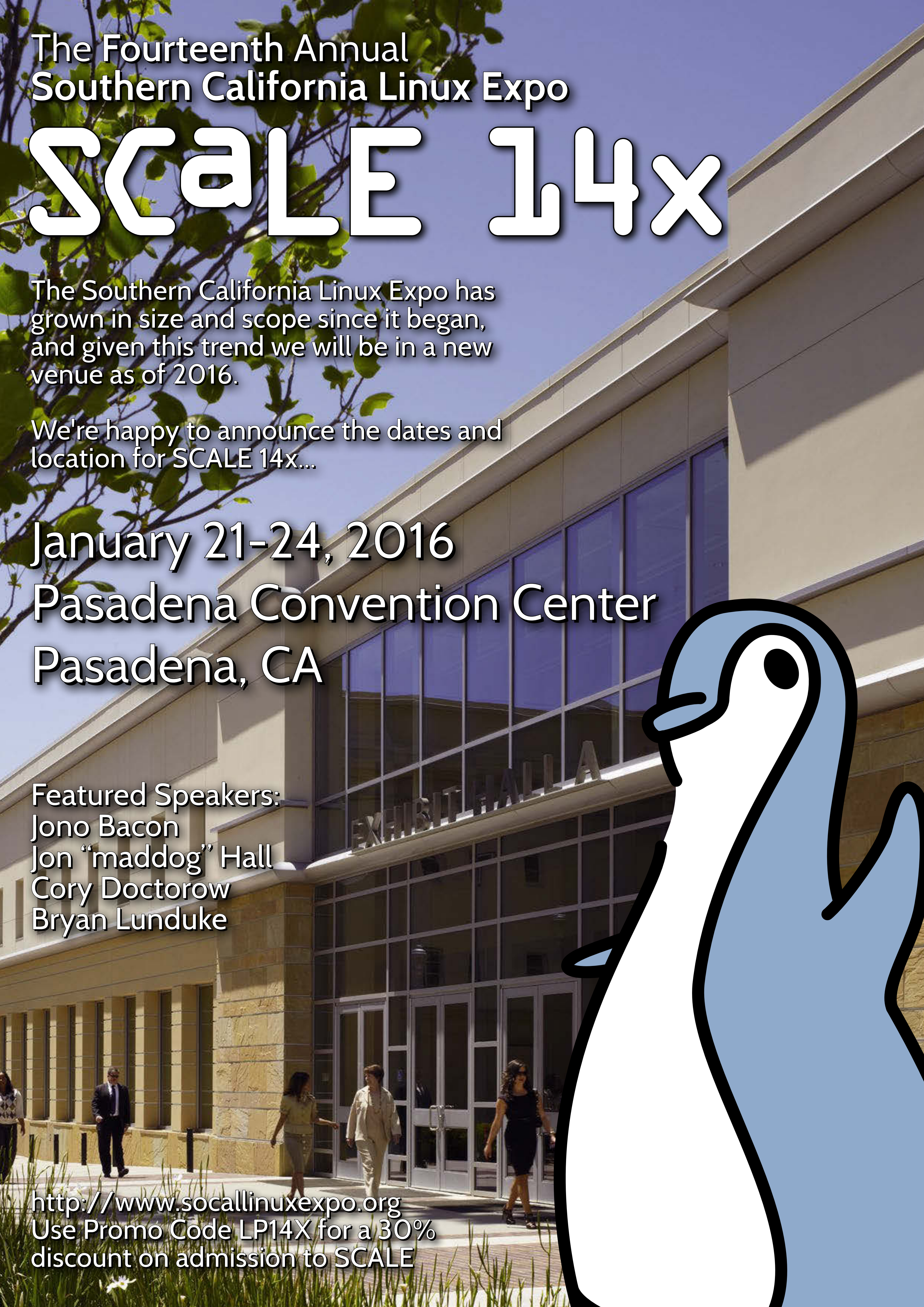


EXHIBIT HALL A

NEWS

Updates on technologies, trends, and tools

THIS MONTH'S NEWS

Firefox Finally Gets Tracking Protection

The Mozilla foundation has announced the arrival of Firefox 42 for Windows, Mac, Linux, and Android. The arrival of a new Firefox is hardly big news (the project releases a new version approximately every six weeks), but number 42 is significant for including a feature that has long been advocated (and debated) in the Firefox community: built-in protection from ad tracking.

According to the Firefox team, "We first added Private Browsing to Firefox to give you control over your privacy locally by not saving your browser history and cookies when you close a private window. However, when you browse the Web, you can unknowingly share information about yourself with third parties that are separate from the site you're actually visiting, even in Private Browsing mode on any browser. Until today. Private Browsing with Tracking Protection in Firefox for Windows, Mac, Android and Linux actively blocks content like ads, analytics trackers, and social share buttons that may record your behavior without your knowledge across sites."

Another interesting new feature of the latest release is tab audio indicators, an audio mini-control that is visible on the tab, so you can change the volume or mute the sound for a web page without placing the page in the foreground.

See the blog post by Mozilla VP Nick Nguyen for more on Firefox 42.

New Password Looter Tool Steals Passwords from a Password Manager

If attackers are on your system, saving your passwords in a password vault is no protection. A new application called Keefarce, which was created by New Zealand developer Denis Andzakovic, steals passwords from the KeePass password manager tool. An attacker who gains access to a system could use Keefarce to output all the user's online passwords to an easily accessible file.

Keefarce does not attack the KeePass encryption system directly but, instead, uses DLL injection to get the KeePass application to export usernames and passwords to a cleartext CSV file. The attack lifts the passwords out of memory in as intended to run when the user has logged in and "unlocked" the password manager.

As the story in Ars Technica points out, KeePass developers have long warned that no password manager is safe when the system itself is compromised. Tools such as key loggers have been harvesting passwords for years on compromised systems. The distinguishing feature of Keefarce is its convenience – you can scoop up all the user's passwords at once.

Experts point out that KeePass is by no means the only password manager tool that is susceptible to this kind of attack, and many believe that storing your passwords in a password vault is still a good policy if it allows you to maintain more unique and less-crackable passwords – just make sure you don't get owned.

Over a Million Websites Are Still Using SHA-1

Faulty hash algorithm persists, despite efforts by experts to raise awareness. A study by the security firm Netcraft has determined that more than a million websites are still using SSL certificates based on the SHA-1 hashing algorithm, which is known to be insecure. Several high-profile companies are among the list of organizations that still use the discredited SHA-1.

Security experts have known for a few years that SHA-1 is vulnerable to attack, with the only question being how much does it cost to attack it? According to a report in the Register, in 2012, it was estimated that a successful attack on SHA-1 would cost $173,000 in compute time by 2017. Netcraft reports the attack can now be accomplished with $75-$120K in Amazon EC2 compute resources.

©Rgbspace, Fotolia.com

Although such as rate would rule out script kiddies and various small-time hackers, a $75,000 investment to hack a corporate network is well within the budget of many criminal and government espionage organizations.

All networks are strongly advised to upgrade to certificates based on SHA-2 and SHA-3-family algorithms.

Ubuntu 15.10 "Wily Werewolf" Appears

Focus is on cloud and OpenStack as Canonical plugs away at the Unity desktop and adds the 4.2 kernel

Canonical developer Adam Conrad has announced the release of Ubuntu 15.10 "Wily Werewolf." The latest release is the first to include a Linux 4.2-based kernel and the gcc-5 compiler collection. According to the announcement, the Ubuntu desktop edition includes "incremental improvements," such as newer versions of GTK and Qt, Firefox, LibreOffice, and the Unity desktop.

The server edition places the emphasis on OpenStack, with support for the latest OpenStack Liberty release and a full complement of OpenStack modules. Other improvements include new powers for the Juju orchestration tool, Open vSwitch 2.4.0, and the Ceph 0.94.3 "Hammer" distributed storage system.

Ubuntu provides separate editions for various IT use cases, including Desktop and Server, as well as a Cloud edition and the Snappy Core version for embedded single-board systems and devices. The Ubuntu team also supports an entourage of related projects built around different desktops and toolsets. Appearing along with the main Unity-based Ubuntu release were new versions of Kubuntu, Lubuntu, Ubuntu GNOME, Kylin (for Chinese-language users), MATE, Ubuntu Studio, and Xubuntu. See the Ubuntu 15.10 release notes for more on the latest version.

Ubuntu 15.10 is a standard release, with nine months of free security updates for desktop and server users. The first release of 2016, Ubuntu 16.04 "Xenial Xerus," will be another Long-Term Service (LTS) release, with five years of bug fixes and security updates for both the server and desktop editions.

MORE ONLINE

Linux Pro Magazine

www.linuxpromagazine.com

Off the Beat • Bruce Byfield

The Seduction of the New

Christiann MacAuley's cartoon, "An Upgrade Is Available for Your Computer" has been making the uncredited rounds of social media sites for the last few years. The cartoon shows the reactions of users on different computers to the news of an upgrade: The Linux user is enthusiastic, the Windows user groans, and the Mac user is glad it will only cost him $99.

Open FOSS Training Needs Donations

Technical documentation was my bridge from academia to journalism and remains a concern of mine. Free software frequently lacks documentation, and even more frequently, it lacks documentation for complete beginners.

Linus Torvalds and the Three Stages of Celebrity

No matter what you think of the accusations that Linus Torvalds encourages a culture of abuse in the kernel project, one thing is clear: he's well into the second level of celebrity, which complicates the situation immensely.

Productivity Sauce • Dmitri Popov

Use Chromium and Chrome Browsers as Timers for the Pomodoro Technique

To practice the Pomodoro technique, you need one thing: a timer. And there are plenty of timer tools and applets out there designed specifically for the Pomodoro technique. If you happen to use Google Chrome or Chromium as your preferred browser, you can use the set timer for search query to activate a timer in a separate window or tab.

Krill: News Filtered

Overwhelmed by the news stream in your regular RSS aggregator? Try Krill. It may look like yet another text-based RSS aggregator, but this nifty little application has a few clever tricks up its sleeve. For starters, Krill can handle not only RSS and Atom feeds, but also Twitter.

DIY GPS Tracking for Your Android Device

Being able to pinpoint the current position of your Android device can come in useful in many situations, but using a third-party app or service may not appeal to privacy-conscious users. If you are one of them, Self-Hosted GPS Tracker might be right up your alley.

MORE ONLINE

ADMIN Online

http://www.admin-magazine.com/

OpenSMTPD Makes Mail Server Configuration Easy • Tobias Eggendorfer
The OpenBSD origins of the OpenSMTPD mail transfer agent makes SMTP easier to implement and manage and more secure.

Improving Docker Security Now and in the Future • Sebastian Meyer
The focus for container solutions such as Docker is increasingly shifting to security. Some vulnerabilities have been addressed, with plans to take further steps in the future to secure container virtualization.

Analyzing Large Volumes of Data with Apache Storm • Holger Reibold
We take you through the installation of a Storm cluster and discuss how to create your own topologies.

Managing Networks in Windows Server vNext • Thomas Joos
We look at a new component in Windows Server vNext – the Network Controller server role.

Successful Protocol Analysis in Modern Network Structures • Matthias Hein
Virtual networks and server structures require additional mechanisms to ensure visibility of data streams. We show how to monitor and analyze network functions, even when virtualization is involved.

Time Protocol Threat Could Allow Login with Expired Passwords

Timely warning sheds new light on problems with the ubiquitous Network Time Protocol.

Cisco's Talos threat intelligence service has uncovered a flaw in the Network Time Protocol (NTP) authentication process that lets an attacker force the NTP daemon into pairing with a malicious time source. According to Talos, this attack "… leverages a logic error in ntpd's handling of certain crypto-NAK packets. When a vulnerable ntpd receives an NTP symmetric active crypto-NAK packet, it will peer with the sender, bypassing authentication typically required to establish a peer association."

Although a time protocol does not provide direct access to financial or medical information, an attacker can do considerable damage if allowed to manipulate network time. Some network services will fail if the system time is out of sync, and control over time parameters could allow access through expired passwords or certificates. Attackers could also cover their tracks or manipulate banking transactions by surreptitiously altering timestamps.

Users are advised to upgrade to ntp-4.2.8p4, which fixes this vulnerability. If an upgrade isn't possible at this time, the Talos report describes some tips for firewall rules that could help mitigate the problem.

Not Another Flash Zero-Day Exploit!

Even patched versions of the porous multimedia tool succumb in the latest attack.

Researchers at Trend Micro say they have discovered yet another zero-day Adobe Flash exploit. The once-popular Flash technology has been the subject of several serious attacks in the past year, and some browsers are actually starting to disable it by default because of security issues.

The latest attack, by the espionage group Pawn Storm, targets government foreign affairs offices around the world and appears to have been launched through email phishing messages.

Dell Pays $67 Billion for EMC

Huge purchase will help Dell face off with huge competitors like Microsoft, HP, and IBM.

Dell has announced that it is buying the storage and enterprise technology giant EMC. The $67 billion price tag is considered the largest tech purchase in history. According to the press announcement, "The combination of Dell and EMC will create the world's largest privately controlled, integrated technology company …. The transaction combines two of the world's greatest technology franchises with leadership positions in servers, storage, virtualization and PCs, and it brings together strong capabilities in the fastest growing areas of the industry, including digital transformation, software-defined data center, hybrid cloud, converged infrastructure, mobile, and security."

Dell got its start selling home and small office PCs, but hardware vendors have known for years the real money is in corporate contracts with enterprise clients. The company has succeeded in bringing itself into the enterprise space, but it is behind some of its larger competitors in recent technologies such as virtualization, private cloud, and Big Data-style storage solutions. This deal should keep them in the conversation with competitors such as Microsoft, Oracle, IBM, and HP.

Some experts, however, are baffled by the announcement and warning of risks associated with combining two such large and disconnected companies. The biggest prize in the EMC portfolio is the popular VMware virtualization solution and its surrounding technologies. VMware will fit well into the pitch Dell needs to make with large enterprise clients.

Front end, back end, dev ops, coding, freelance, part time, and full time: whether you work in an office or work from home Drupal Jobs has it all.

No tricks and no spam -- just great employers looking for Drupal talent.

When you find work on Drupal Jobs, you give back to the community.

It’s time to get to work.

Drupal™ Jobs

Drupal Jobs is the official job board of the Drupal community
and is run by the Drupal Association

Custom solutions for system monitoring and control

Just Right

Off-the-rack monitoring tools often offer too many functions or fail to offer precisely what you need, but shell scripts let you create individual monitoring routines.

By Harald Zisler

Trust is good, but keeping the thumb screws on is better: This is the principle by which IT services and functions are monitored. Although you can find many tools to accomplish this job, tailor-made monitoring doesn't actually need these giants. Simple shell scripts will take you where you need to go just as well.

Whether you need to monitor and control a web server, database system, network connections, users, fans, or computer temperatures, simple shell routines are typically reliable and fast. Once created, scripts can be modified for different distributions and scenarios.

Monitoring needs to be considered carefully, however: In the case of monitoring a web server, it is not just a question of checking that the service is running – the question lacks precision. Is the hardware running? To determine this, all you need is a simple `ping`. A positive response, however, by no means signifies that the web server daemon is working. To discover this, you need to query the process status locally on the server; that is,

```
ps -C <service>
```

or possibly

```
service <service> status
```

However, you still don't know whether users can retrieve data from the web server. You would need to test this regularly in a browser, preferably in an automated process using a command-line tool and ideally from somewhere outside of your own protected network infrastructure. Otherwise, you risk being lulled into a false sense of security – for example, even when a router no longer works.

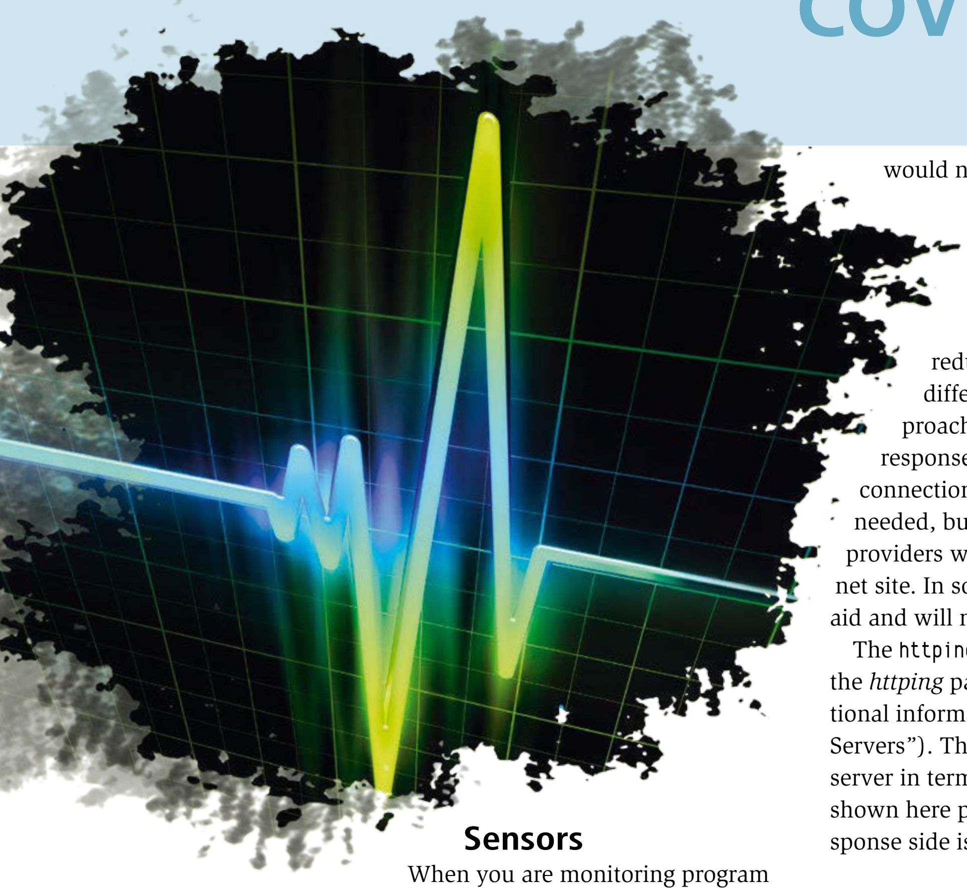
TABLE 1: Test Tools

Test Objective	Tool
Accessibility of websites	`httping` (1)
Database shell client for PostgreSQL-RDBMS	`psql`
Accessibility of computers	`ping`
Name resolution	`host`
Logged on users	`users`
Service status (SysVinit)	`/etc/init.d/<Service> status`
Service status (Systemd)	`systemctl status <Service>`
Disk space	`df`
Temperature	`sensors`
Fan activity	`sensors` (2)
Port access	`netread` (3)

Packages: (1) httping, (2) lm-sensors, (3) netrw

AUTHOR

Harald Zisler has focused on FreeBSD and Linux since the early 1990s. He is the author of various articles and books on technology and IT topics. The third edition of his book *Computer-Netzwerke* (Computer Networks) was recently published by the Rheinwerk Verlag publishing company. He also works as an instructor, teaching Linux and database topics in small groups.



Sensors

When you are monitoring program execution, the task is to check the exit codes that terminal applications and commands typically output after terminating – gracefully or not. A value of `0` typically signals a successful program run, whereas other codes indicate more or less serious errors. Table 1 provides a brief selection of popular tools for system monitoring.

As the example of monitoring a web server shows, monitoring involves a little overhead in some cases (Figure 1). In this case, monitoring would ideally not be operated in-house but from outside of your own IT infrastructure so that failures would not also take down the monitoring system. In this way, you can cover almost all failure cases: web lockouts, overloaded attacks, general network overload, and even cases of physical network disconnection – think backhoes.

In response, you could (automatically) fire up a redundant system at some other location or with a different Internet connection. Listing 1 shows an approach that also clears up other questions as an initial response to delimiting an error (DNS problem, network connection, and more). This script can be extended easily if needed, but watch out for pitfalls caused by some Internet providers when you attempt to access an unreachable Internet site. In some cases, you will be shown a helpful navigation aid and will not want to evaluate the HTTP status there.

The `httping` command executed on the script (typically from the *httping* package) calls the stated website and displays additional information, such as latency (see the box "Pinging Web Servers"). This means you can quite easily monitor a web server in terms of functionality. The system monitoring script shown here provides the sensor system for monitoring; the response side is typically outsourced into a second script.

PINGING WEB SERVERS

The httping program checks access to a web server; it can optionally also determine the response behavior, assuming the connection is not routed via a proxy server or does not transfer the complete page content using the `-G` option, which would falsify response times. The basic call uses the syntax

```
httping -g <URL>
```

and you can use the `-p <port>` option to stipulate a port other than the typical port 80.

If so desired, httping will generate helpful information on top of the exit codes (`0` = functioning, `127` = error), including the response time, which assumes a value of `-1` for an error. By passing in a variable, you can trigger alarms or responses based on these results. For a better understanding of the function, launch the small sample script from Listing 2. Listing 3 shows the matching output.

The first call targets a working website. Httping shows the response time and the HTTP status code `200`. If you point httping at a working domain, but a non-existent website, the test tool will output the classical 404 error with a response time of `-1`. If the domain doesn't exist, then the Internet provider in this example redirects the script to its own navigation aid with an integrated search function; therefore, httping does not report *Resolving exshample.com failed* but outputs `302` – the status code for redirection.

Monitoring Databases

Databases are another important building block in any IT infrastructure, and it is obviously important to monitor them. The possibilities include MySQL, MariaDB, or PostgreSQL data-

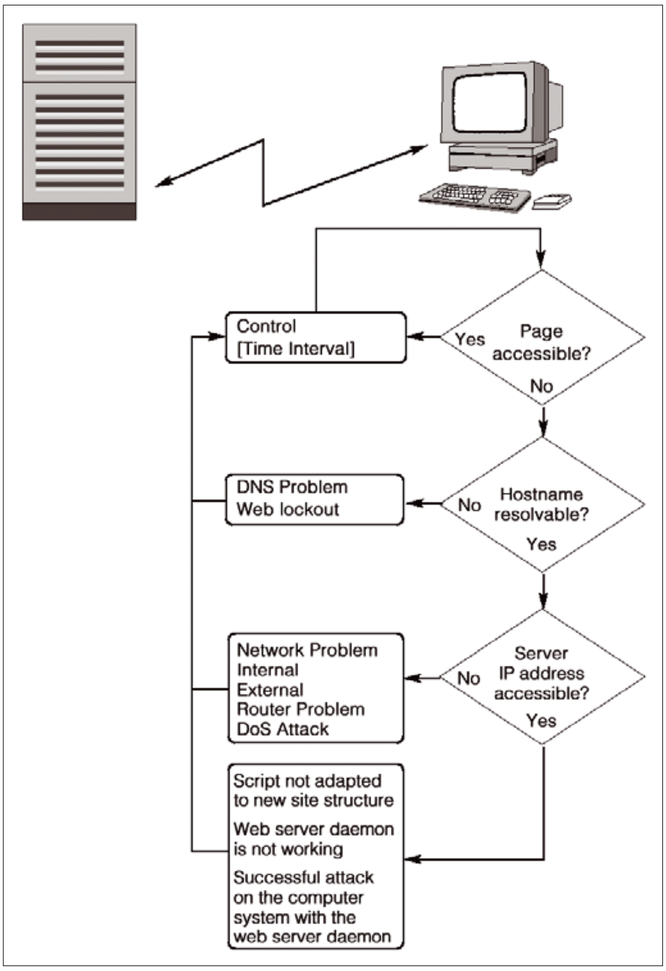
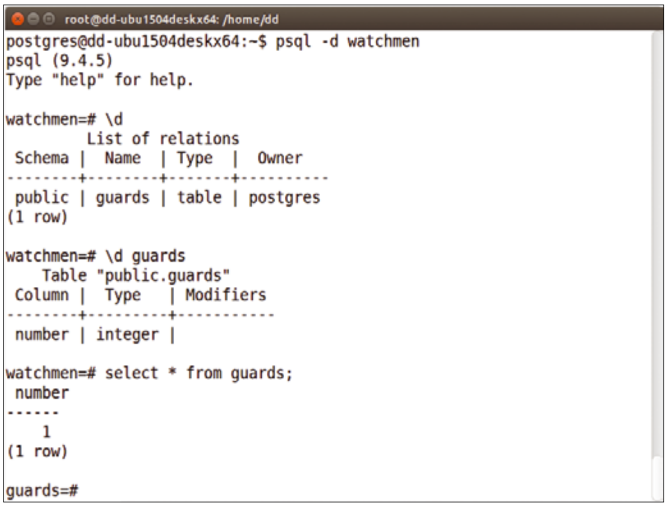

Control
[Time Interval]
Yes
Page accessible?
No
DNS Problem
Web lockout
No
Hostname resolvable?
Yes
Network Problem
Internal
External
Router Problem
DoS Attack
No
Server IP address accessible?
Yes
Script not adapted to new site structure
Web server daemon is not working
Successful attack on the computer system with the web server daemon

Figure 1: Schematic sequence of web server functional monitoring.



```
root@dd-ubu1504deskx64: /home/dd
postgres@dd-ubu1504deskx64:~$ psql -d watchmen
psql (9.4.5)
Type "help" for help.

watchmen=# \d
        List of relations
 Schema |  Name  | Type  |  Owner
--------+--------+-------+----------
 public | guards | table | postgres
(1 row)

watchmen=# \d guards
    Table "public.guards"
 Column |  Type   | Modifiers
--------+---------+-----------
 number | integer |

watchmen=# select * from guards;
 number
--------
      1
(1 row)

guards=#
```

Figure 2: A simple database to check the functionality of a PostgreSQL server.

bases: I focus on PostgreSQL in this example. To monitor the service, you need to create a separate user account and a database with the table for this account. In doing so, the shell scripts can automate the query process. In this example, the database is named `watchmen`, and it contains the `guards` table with a `number` column and a single record (Figure 2).

The `psql` shell client uses classical exit codes: `0` for okay and 1 for failed. The shell script in Listing 4 then decides whether the data is simply inaccessible for some reason or whether the service is not working at all. Figure 3 shows the procedure. For test purposes, I deleted the data on one occasion and stopped the service on another. Assuming that the script is running on the same computer as the relational database management system, you can also perform other actions.

Monitoring Services

Many services simply work away in the background, and you are unable to talk to them directly through a web or database

LISTING 1: Web Server Monitoring

```
#! /bin/sh
HOST=www.example.com
IP=93.184.216.34

while true; do

  # Access website, output to variable
  B=$(httping -G -g $HOST -c 1 -s -m)
  # Store exit code in variable
  A=$?

  # Break down httping output
  C=$(echo $B  | cut -d \  -f1)
  D=$(echo $B  | cut -d \  -f2)

  # Output variables
  echo "Exit-Code: $A"
  echo "STATUS: $C"

  # Check name resolution
  if [ "$C" = "-1" ]; then
    host $HOST
    # Store exit code ...
    NA=$?
    # ... and evaluate
    if [ $NA = 0 ]; then
      echo "Name resolution ok"
    else
      echo "Name resolution error"
      # Availability via IP address?
      ping -c 1 -q $IP
      # Store exit code ...
      E=$?
      # ... and evaluate
      if [ $E -eq 0 ]; then
        echo "Computer accessible on network"
      else
        echo "Computer not accessible on network"
      fi
    fi
  fi

  # Note, if page can be retrieved
  if [ $D -ne 200 ]; then
    echo "Page error $D"
  fi

  sleep 15

done
```

LISTING 2: Website Test Script

```
#! /bin/sh
echo "This website works:"
httping -g http://www.example.com -c 1 -s -m
echo "-----------------------------------------------"
echo "Domain exists, but invalid page:"
httping -g http://example.com/page-not-there.html -c 1 -s -m
echo "-----------------------------------------------"
echo "Domain does not exist, redirected by provider:"
httping -g http://exshample.com -c 1 -s -m
```

LISTING 3: Script Output

```
$ ./listing2.sh
This website works:
206,761122 200
-----------------------------------------------
Domain exists, but invalid page:
-1 404
-----------------------------------------------
Domain does not exist, redirected by provider:
-1 302
```

NO LIMIT HOSTING

*Server with dedicated **1 Gbit/s port** and **unlimited** traffic!*

Unlimited fixed-price hosting

No risk traffic flatrates:

Volume Plus 500

- 500 Mbit/s
- unlimited traffic
- performance boost

$**0** /month

Volume Plus 1000

- 1,000 Mbit/s
- unlimited traffic
- maximum performance

$**550** /month

Our **most popular** server:

NoLimit **Storm**

- HP DL120 G7
- Intel-Xeon E3-1270
- 16 GB DDR3 RAM and 2×2 TB SATA II

No setup fee!

Monthly contract period

$**109** /month

All prices excluding VAT. Our offers are for commerce and industry.

Dedicated **1Gbit/s** port included

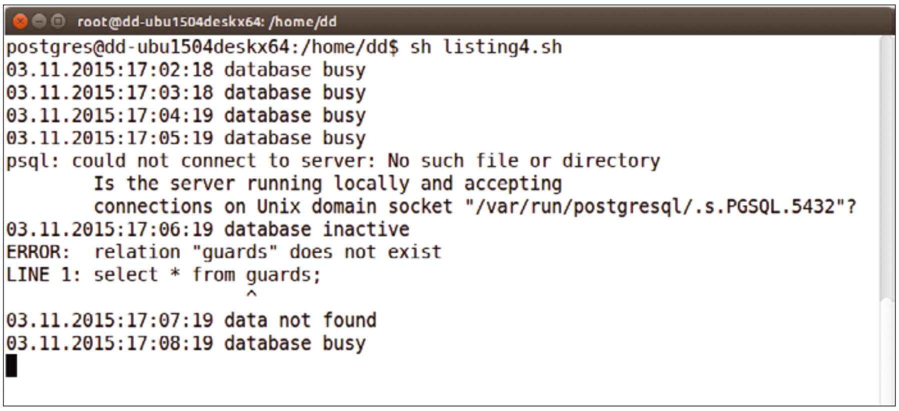
Call us for more details!
+1 314 – 4 92 46 16

UNMETERED.COM
NO LIMIT HOSTING

server; thus, it is impossible to check the availability of this kind of service with a simple query. In these cases, you need to rely on the service fulfilling its task if its process is active.

Status queries can be made based on the examples in Table 2. The simplest tool for this task is the `ps` command. The `-C <process>` option lets you restrict the search for the process in question to the stipulated name (Listing 5).

The exit codes returned by `ps` can be processed easily in scripts further downstream. Alternatively, you can pick up the output from the init scripts called by the `service` command (Listing 6) or, for distributions with systemd, by the `systemctl` command (Listing 7).



```
postgres@dd-ubu1504deskx64:/home/dd$ sh listing4.sh
03.11.2015:17:02:18 database busy
03.11.2015:17:03:18 database busy
03.11.2015:17:04:19 database busy
03.11.2015:17:05:19 database busy
psql: could not connect to server: No such file or directory
        Is the server running locally and accepting
        connections on Unix domain socket "/var/run/postgresql/.s.PGSQL.5432"?
03.11.2015:17:06:19 database inactive
ERROR:  relation "guards" does not exist
LINE 1: select * from guards;
                      ^

03.11.2015:17:07:19 data not found
03.11.2015:17:08:19 database busy
```

Figure 3: Checking the functionality of a PostgreSQL server.

Whereas the legacy SysVinit forces you to evaluate the output from the init scripts that you call, which involves considerable overhead, systemd returns more useful exit codes. Depending on the task, requirements, and the system, you can use one of the three methods introduced here to achieve your objectives when monitoring a service.

LISTING 4: Checking PostgreSQL Server

```
#! /bin/sh
while true; do
  # Write date and time to variable
  TIME=$(date +%d.%m.%Y:%H:%M:%S)
  # Database query to extract the exit code
  M=$(psql -q -d watchmen -c "select * from guards;")
  # Store and evaluate exit code
  A=$?
  if [ $A -eq 0 ]; then
    echo "$TIME Database working"
  elif [ $A -eq 1 ]; then
    echo "$TIME Data not found"
  elif [ $A -eq 2 ]; then
    echo "$TIME Database inactive
  fi
  sleep 60
done
```

LISTING 5: Status Query for NTP Daemon

```
$ ps -C ntpd
  PID TTY          TIME CMD
 1054 ?        00:00:01 ntpd
$ echo $?
0
# Stop service, with system in this case
$ sudo systemctl stop ntpd.service
$ ps -C ntpd
  PID TTY          TIME CMD
$ echo $?
1
```

LISTING 6: Service (SysVinit)

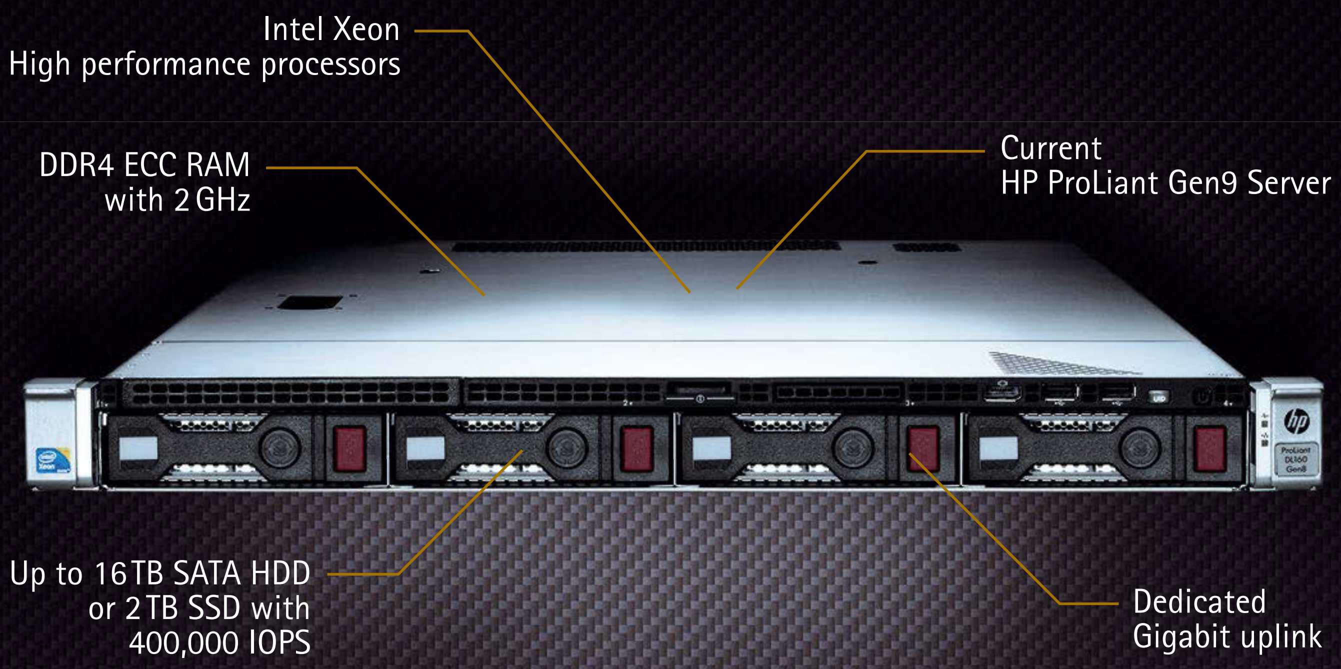
```
$ service ntp status
 * NTP server is running
$ sudo service ntp stop
 * Stopping NTP server ntpd                  [ OK ]
$ service status
 * NTP server is not running
```

LISTING 7: Systemctl (systemd)

```
$ systemctl status ntp
|- ntp.service - LSB: Start NTP daemon
   Loaded: loaded (/etc/init.d/ntp)
   Active: active (running) since Mon 2015-10-26 19:22:03
           CET; 43s ago
[...]
$ echo $?
0
$ sudo systemctl stop ntp
$ systemctl status ntp
|- ntp.service - LSB: Start NTP daemon
   Loaded: loaded (/etc/init.d/ntp)
   Active: inactive (dead) since Mon 2015-10-26 19:23:04
           CET; 4s ago
[...]
$ echo $?
3
```

TABLE 2: Service Check

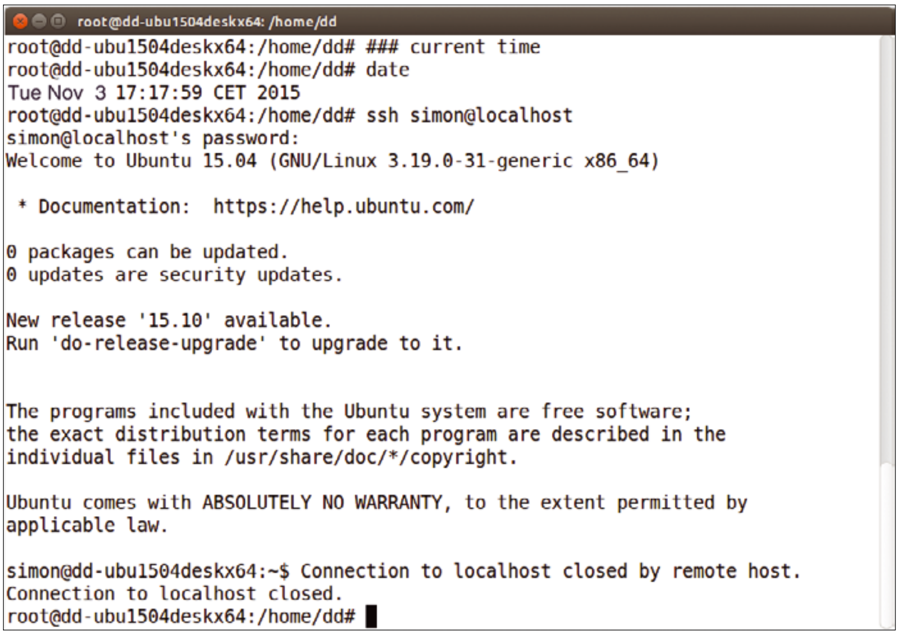
Method	Call	Exit Code
Process status, stating the service	`ps -C <process>`	`0` = exists; `1` = does not exist
Init script with option `status`	`service <Start script> status`	None; outputs individual messages instead
Query with systemctl	`systemctl status <service>`	`0` = active, `3` = deactivated
Communication with the service	–	See examples of web and database servers


Intel Xeon
High performance processors
DDR4 ECC RAM
with 2 GHz
Current
HP ProLiant Gen9 Server
Up to 16 TB SATA HDD
or 2 TB SSD with
400,000 IOPS
Dedicated
Gigabit uplink

Reacting to Logins

From these building blocks, you can create small scripts with easy tools that respond to events on your system. For example, you can quite easily use logins to trigger actions – for example, to avoid discussions with the kids every evening if they have been sitting in front of the computer too long. When the bell tolls, you can then log them off the system automatically whether they are happy with it or not.

The script in Listing 8 relies on `users` to discover the users currently logged on to the system and then on `grep` to filter the output. This gives you an exit code of `0` in case of a match and of 1 in case of a miss. Based on the results, and the current time (assuming the hours between 21:00 and 07:00 are off limits), the script then allows user *simon* to use the system or throws him out without so much as a by your leave (Figure 4).



```
root@dd-ubu1504deskx64: /home/dd
root@dd-ubu1504deskx64:/home/dd# ### current time
root@dd-ubu1504deskx64:/home/dd# date
Tue Nov  3 17:17:59 CET 2015
root@dd-ubu1504deskx64:/home/dd# ssh simon@localhost
simon@localhost's password:
Welcome to Ubuntu 15.04 (GNU/Linux 3.19.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

0 packages can be updated.
0 updates are security updates.

New release '15.10' available.
Run 'do-release-upgrade' to upgrade to it.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

simon@dd-ubu1504deskx64:~$ Connection to localhost closed by remote host.
Connection to localhost closed.
root@dd-ubu1504deskx64:/home/dd# █
```

Figure 4: Automatically logging off the user if they exceed their time window.

The principle can also be applied to convenience and automation functions. For example, with a slightly modified script (Listing 9) you can launch a web server as soon as a specific user (*jefe* in this example) logs on. In this case, the script is designed for use with a systemd computer; you might also need to change the name for the web server service (in this example, *httpd.service*).

The user does not need to enter any commands or initiate any actions, and you don't even need to make any changes to home directories or user accounts. For example, in `~/.profile` or `~/.bashrc`), all you need is an active session. This approach lets you assign important tasks, such as starting a service or a backup, to a non-privileged user account. Of course, you need to call the shell script automatically when the computer boots. The approach is different, depending on the init system.

LISTING 8: Automatic Logoff

```
#! /bin/sh
while true; do
  # Discover the time
  TIMENOW=$(date +%H)
  # Is Simon logged on?
  users | grep -q simon
  # Evaluate exit code
  A=$?
  if [ $A -eq 0 ]; then
    # Simon is logged on
    if [ $TIMENOW -le 7 ]  || [ $TIMENOW -ge 21 ]; then
    # before 07:00 or after 21:00 hours
      killall -u simon
    fi
  fi
  sleep 60
done
```

As a general rule, you would want to store the executable shell script in the `/usr/local/sbin` directory. For older systems, such as Debian 7 with SysVinit, you would typically copy the `skeleton` template, which is already in place on

LISTING 9: Launching a Web Server Conditionally

```
#! /bin/sh
while true; do
  # Is jefe logged on?
  users | grep -q jefe
  # Evaluate exit code
  A=$?
  if [ $A -eq 0 ]; then
    # jefe logged on
    # Check status of web server
    systemctl status httpd.service
    # Evaluate exit code
    B=$?
    # Start web server if needed
    if [ $B -gt 0 ]; then
      systemctl start httpd.service
    fi
  else
    # jefe not logged on
    # Check status of web server
    systemctl status httpd.service
    # Evaluate exit code
    C=$?
    if [ $C -eq 0 ]; then
      # Stop web server if needed
      systemctl stop httpd.service
    fi
  fi
  sleep 60
done
```

most systems, to a new file with an appropriate name below /etc/init.d (this example uses weblogon). Next, edit the init script to suit your needs, save the changes, and make the file executable by running chmod +x. To enable the script for the typical runlevel, enter:

```
update-rc.d weblogon defaults
```

If you work on a computer with systemd, you need to create a new unit file named weblogon.service, as shown in Listing 10, under the /usr/local/lib/systemd/system directory, which you might have to create, and run the code in Listing 11 to link the instructions with the system and ensure an automatic start at bootup.

Messages via Email

Optionally, you can also tell your computer to notify you by email in case of important events. To do so, you need a mail transport agent (MTA), such as SSMTP [1] on your computer. The agent forwards email via a regular SMTP server so that the server mail is not discarded automatically as spam by the receiving email service.

Listing 12 shows a simple shell script that notifies you of all users still logged on to the computer after 22:00 hours. To handle this task, the script creates a helper file named mail.txt, which ssmtp then uses as input for the email to be sent.

Before starting the script, you first need to set up SSMTP. To do so, use the configuration files ssmtp.conf (Listing 13) and revaliases (Listing 14) from the /etc/ssmtp directory. Depending on which email provider you want to address, you need to enter different configuration details at this point. If you are having difficulty finding the right information, searching the Internet with *<Provider> ssmtp* will typically help.

The examples in the configuration files are applicable for a normal FreeMail provider. For test purposes, you can run ssmtp in the shell script with the -v option. The program is then far more verbose, allowing you to discover problems with the transmission more quickly and make the necessary changes.

Make sure that only root, or the user in whose context SSMTP runs, can see the SSMTP configuration file. Check the documentation of your Linux distribution for instructions on doing this (e.g., for Arch Linux [2]).

Conclusions

The examples shown here only cover a fraction of the possibilities that shell scripts offer for system monitoring. To create scripts, you do not need advanced programming capabilities; experience with terminal applications and simple constructs such as while loops and if statements are typically sufficient. ■■■

LISTING 10: weblogon.service

```
[Unit]
Description=Start web server when jefe logs on
Documentation=man:users(1)

[Service]
ExecStart=/usr/local/sbin/webmeldung.sh
IgnoreSIGPIPE=false

[Install]
WantedBy=multi-user.target
```

LISTING 11: Setting Up Autostart at Bootup

```
$ sudo systemctl enable weblogon.service
Created symlink from /etc/systemd/system/multi-user.target.
  wants/weblogon.service to /usr/local/lib/systemd/system/
  weblogon.service.
$ sudo systemctl start weblogon.service
$ sudo systemctl status weblogon.service
* weblogon.service - Start web server when jefe logs on
    Loaded: loaded (/usr/local/lib/systemd/system/weblogon.
            service; enabled; vendor preset: disabled)
    Active: active (running) since Tue 2015-10-27 14:16:13
            CET; 6s ago
[...]
```

LISTING 12: User Notification

```
#!/bin/sh
RECEIVERADDRESS=daddy@example.com
SENDERADDRESS=kidsroom@home.net
while true; do
  timenow=$(date +%H)
  if [ $timenow -eq 22 ]; then
    echo "To: $RECEIVERADDRESS" > mail.txt
    echo "From: $SENDERADDRESS" >> mail.txt
    echo "Subject: User query" >> mail.txt
    echo "" >> mail.txt
    users >> mail.txt
    ssmtp $RECEIVERADDRESS < mail.txt
    sleep 15
    #sleep 3600
  fi
done
```

LISTING 13: ssmtp.conf

```
root=daddy@example.com
mailhub=smtp.example.com:25
hostname=kidsroom
UseTLS=Yes
UseSTARTTLS=YES
AuthUser=<Login for SMTP Server>
AuthPass=<Password for SMTP Server>
FromLineOverride=NO
```

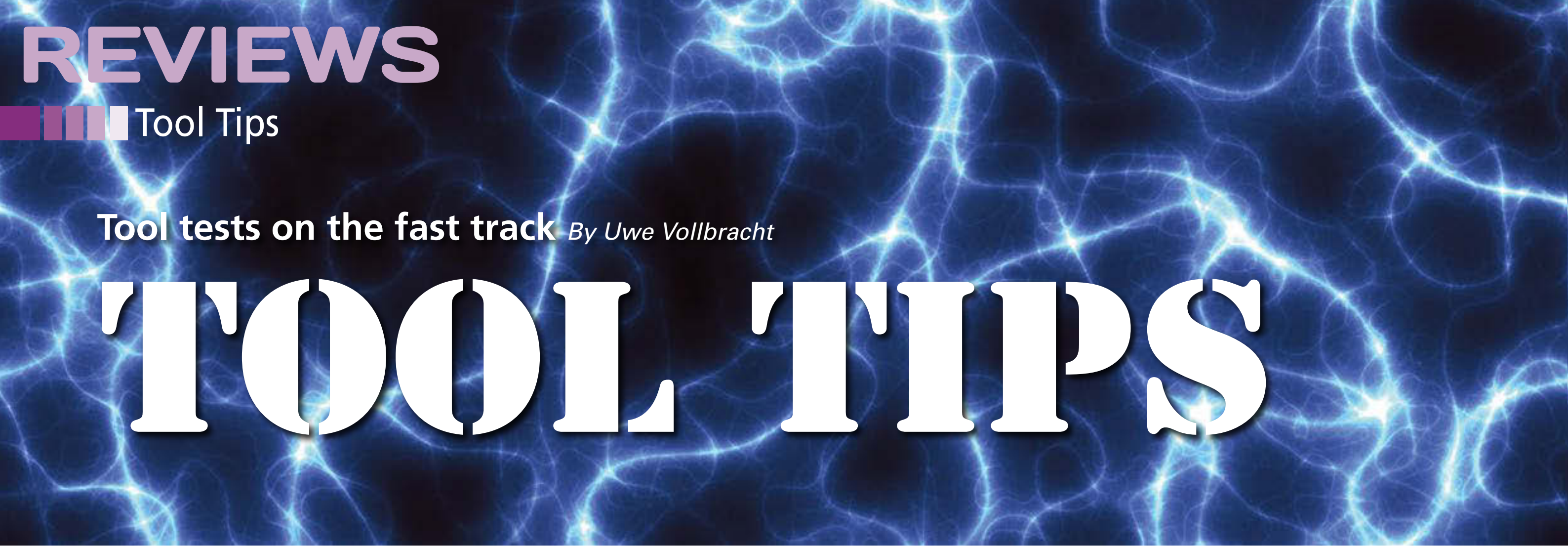
LISTING 14: revaliases

```
root:daddy@example.com:smtp.example.com:25
```

INFO

[1] SSMTP: *https://packages.qa.debian.org/s/ssmtp.html*

[2] SSMTP security: *https://wiki.archlinux.org/index.php/SSMTP#Security*

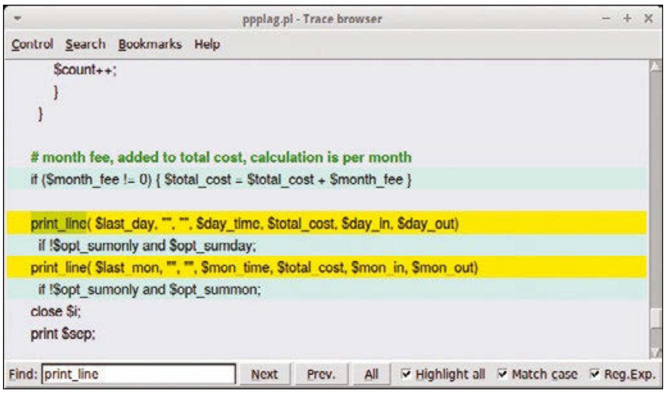


Tool tests on the fast track *By Uwe Vollbracht*

TOOL TIPS

Trowser 1.3

Function: Alternative to Less with highlighting
Source: *http://www.nefkom.net/tomzo/prj/trowser*
License: GPLv3
Alternatives: Less, Xless



```
        $count++;
      }
    }

  # month fee, added to total cost, calculation is per month
  if ($month_fee != 0) { $total_cost = $total_cost + $month_fee }

  print_line( $last_day, "", "", $day_time, $total_cost, $day_in, $day_out)
    if !$opt_sumonly and $opt_sumday;
  print_line( $last_mon, "", "", $mon_time, $total_cost, $mon_in, $mon_out)
    if !$opt_sumonly and $opt_summon;
  close $i;
  print $scp;
```

Trowser is a good alternative to Less with some additional features like color highlighting, search history, and bookmark management. The tool, which is implemented in Tcl/Tk, provides a graphical interface but can also be controlled using Vim shortcuts.

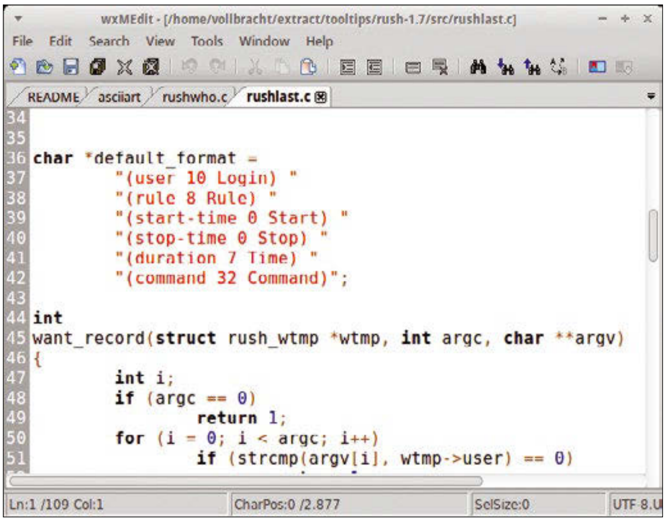
When the program is started for the first time, you need to define highlighting patterns and enter a search term. You can then open the *Edit Highlight Patterns* dialog via the *Search* menu and adjust the color scheme to how you want it. Trowser can highlight both the expression and whole rows. You can also enable case sensitivity and a function for regular expressions by checking the relevant checkboxes. To add a bookmark, you just select a row by double-clicking it; Trowser saves the bookmarks in a separate file with `.bok` as the ending so as not to change the original file.

The developers are planning to expand the filtering options and to manage several searches in sub-windows for future versions.

★★★☆☆ Trowser was impressive in the test. However, learning the keyboard shortcuts might take a while for anyone who hasn't worked with Vim before. ■■■

wxMEdit 2.9.9

Function: Cross-platform editor
Source: *https://wxmedit.github.io*
License: GPLv3
Alternatives: Bluefish, Geany



```


char *default_format =
        "(user 10 Login) "
        "(rule 8 Rule) "
        "(start-time 0 Start) "
        "(stop-time 0 Stop) "
        "(duration 7 Time) "
        "(command 32 Command)";

int
want_record(struct rush_wtmp *wtmp, int argc, char **argv)
{
        int i;
        if (argc == 0)
                return 1;
        for (i = 0; i < argc; i++)
                if (strcmp(argv[i], wtmp->user) == 0)
```

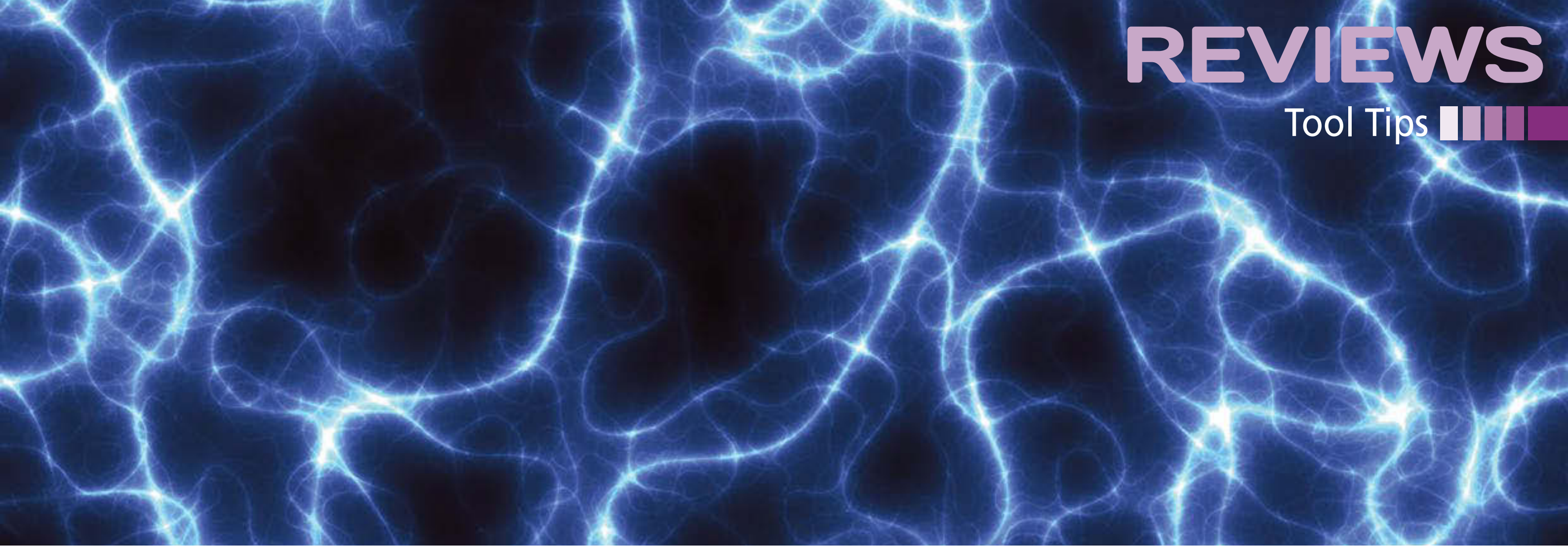
wxMEdit is a cross-platform text and hexadecimal editor that can be used on Microsoft Windows, Linux, FreeBSD, and OS X. The program, which is implemented in C++ and wxWidgets, offers three editing modes – text, column, and hexadecimal – and provides syntax highlighting for more than 30 programming languages.

As well as an automatic update function, the developers equipped the editor with bookmark management, a function for erasing the history, and a context menu for each tab. They also improved integration in Windows and OS X environments, translated the program into several languages, and added new encodings. wxMEdit supports Chinese, Japanese, and Korean characters, musical symbols, and Emoji.

The text editor automatically wraps lines with more than 80 characters. You can adjust this value in the program settings. You can also redefine keyboard shortcuts and color highlighting there.

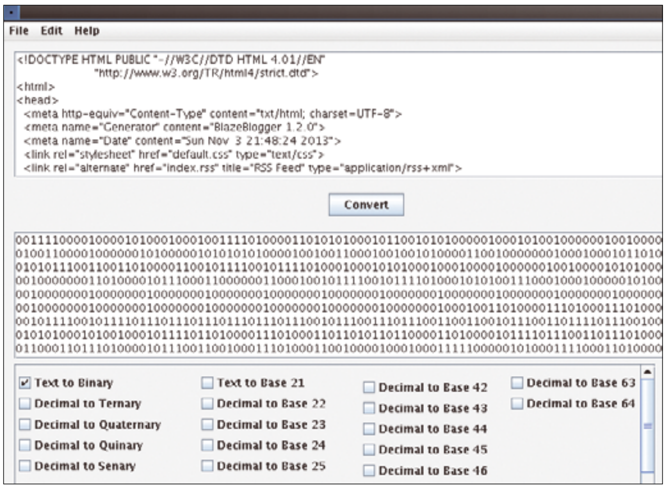
★★★★☆ wxMEdit is a powerful editor that can be adjusted to individual users' needs. It would get full marks if it had interfaces for compilers or external development environments. ■■■

Lead Image © Kheng Ho Toh, 123RF.com



BinaryCrypt 2.0

Function: Convert files and text
Source: *http://miragesoftware.jimdo.com/binarycrypt*
License: GPLv3
Alternatives: none



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
               "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta http-equiv="Content-Type" content="txt/html; charset=UTF-8">
  <meta name="Generator" content="BlazeBlogger 1.2.0">
  <meta name="Date" content="Sun Nov  3 21:48:24 2013">
  <link rel="stylesheet" href="default.css" type="text/css">
  <link rel="alternate" href="index.rss" title="RSS Feed" type="application/rss+xml">
```

```
001111000010000101000100010011110100001101010100010110010101000001000101001000000100100001
010011000010000001010000010101010100001001001100010010010100001100100000001000100010110100
010101110011001101000011001011110010111101000100010101000100010000100000010010000101010000
001000000011010000101110001100000011000100101111001011110100010101001110001000100000101000
001000000010000000100000001000000010000000100000001000000010000000100000001000000010000000
001000000010000000100000001000000010000000100000001000100110100001110100011101000111010000
001011110010111101110111011101110111011100101110011101110011001100101110011011110111001001
010101000101001000101111011010000111010001101101011011000011010000101111011100110111010000
011000110111010000101110011001000111010001100100001000100011111000001010001111000110100000
```

- ☑ Text to Binary
- ☐ Decimal to Ternary
- ☐ Decimal to Quaternary
- ☐ Decimal to Quinary
- ☐ Decimal to Senary
- ☐ Text to Base 21
- ☐ Decimal to Base 22
- ☐ Decimal to Base 23
- ☐ Decimal to Base 24
- ☐ Decimal to Base 25
- ☐ Decimal to Base 42
- ☐ Decimal to Base 43
- ☐ Decimal to Base 44
- ☐ Decimal to Base 45
- ☐ Decimal to Base 46
- ☐ Decimal to Base 63
- ☐ Decimal to Base 64

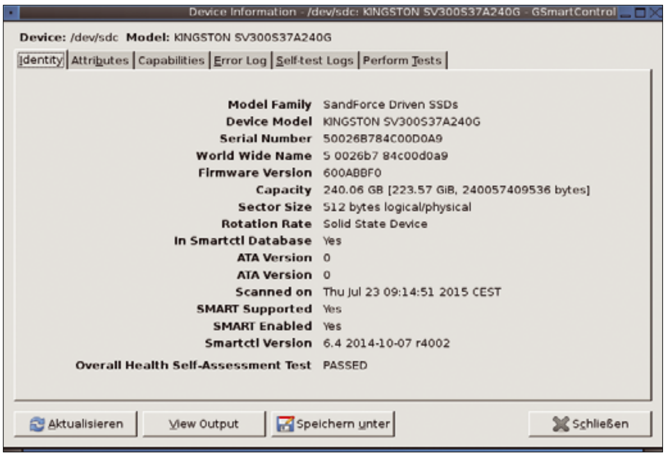
BinaryCrypt converts text into various decimal or binary formats, hexadecimal or octal representations, and back. The tool can handle more than 30 numerical systems, including many decimal formats with a base of 21 to 64.

To use the tool, enter your text in the top field; the program parses longer documents. After you select the target format in the bottom pane, clicking *Convert* will create the result. Depending on your computer's performance, this may take a while with long text. BinaryCrypt takes the *Convert to text* path by default. You can change the approach by choosing *Convert from text*. One of the tool's extra features is a binary computer that performs basic arithmetic operations.

★★★☆☆ The advantages of BinaryCrypt are obvious. The program combines multiple conversion technologies in a single interface. You just need to check the appropriate checkbox to convert text, and you no longer need to use several command-line tools. However, a few features are missing. For example, it's not possible to transmit text in Base64 or decimal entries without taking a detour into binary encoding. ■■■

GSmartControl 0.8.7

Function: Detects hard drive problems
Source: *http://gsmartcontrol.sourceforge.net*
License: GPLv3
Alternatives: Smartctl2



Device: /dev/sdc Model: KINGSTON SV300S37A240G

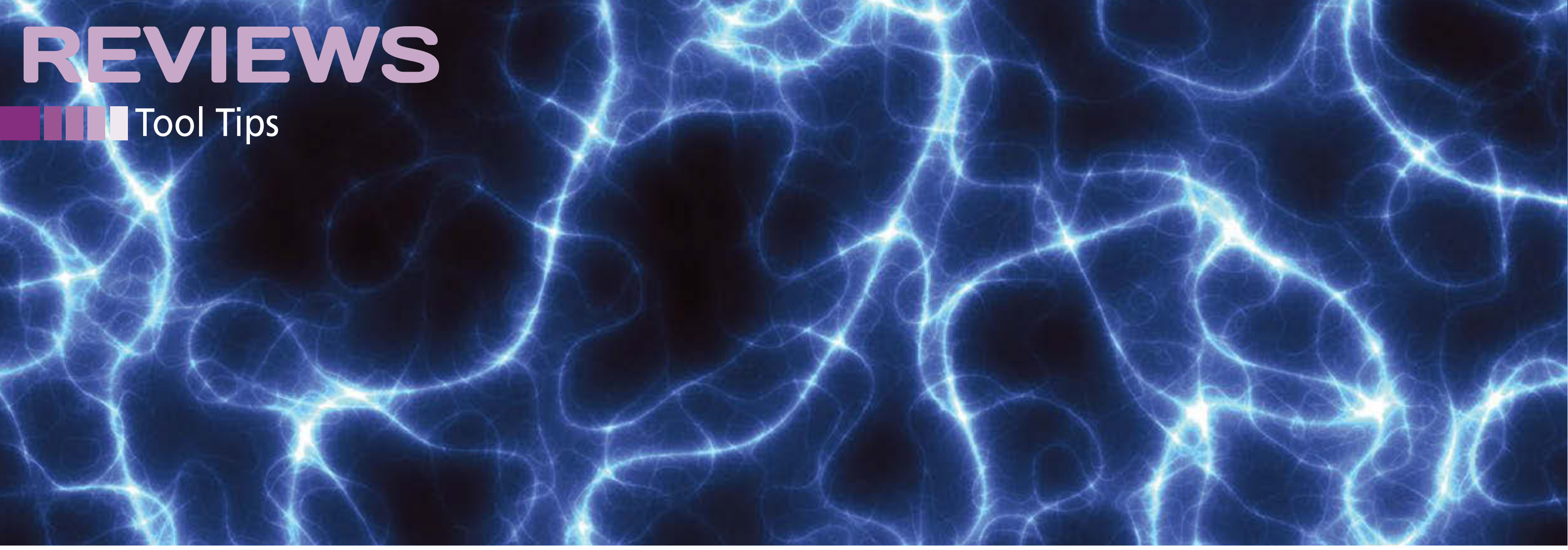
```
                      Model Family  SandForce Driven SSDs
                      Device Model  KINGSTON SV300S37A240G
                     Serial Number  50026B784C00D0A9
                   World Wide Name  5 0026b7 84c00d0a9
                  Firmware Version  600ABBF0
                          Capacity  240.06 GB [223.57 GiB, 240057409536 bytes]
                       Sector Size  512 bytes logical/physical
                     Rotation Rate  Solid State Device
            In Smartctl Database  Yes
                       ATA Version  0
                       ATA Version  0
                        Scanned on  Thu Jul 23 09:14:51 2015 CEST
                   SMART Supported  Yes
                     SMART Enabled  Yes
                  Smartctl Version  6.4 2014-10-07 r4002

Overall Health Self-Assessment Test  PASSED
```

Thanks to S.M.A.R.T. (Self-Monitoring, Analysis, and Reporting Technology), users are made aware of hard disk problems before the data medium vociferously clamors for attention. The hardware collects various information depending on the connection and the type of medium. Linux users, for example, can read this information using the `smartctl` command-line tool. Those who prefer a bit more convenience should use the graphical interface GSmartControl.

The program uses `smartctl` in the background and can therefore handle the same query options for disks and solid-state drives. It identifies connected data media when launched and lists them – regardless of whether they are S.M.A.R.T.-capable. You can disable displaying uncooperative media in the configuration. You can also set individual parameters for each data medium.

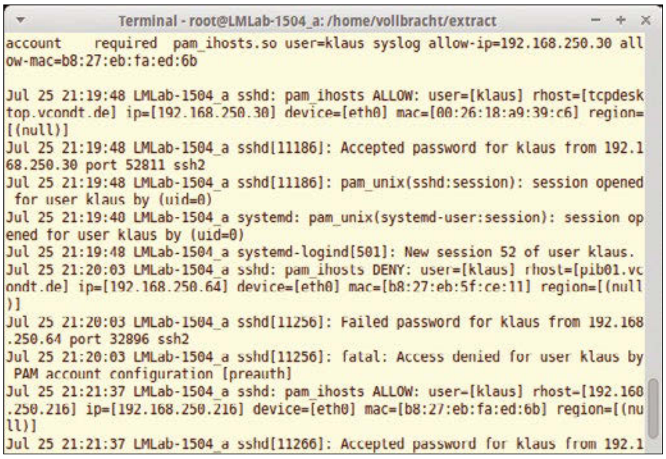
GSmartControl provides information about the medium if you right-click on the corresponding icon. The detailed view has several tabs that you can use to access status indicators and logfiles. The tool highlights errors and threshold values. The *Perform Tests* tab lets you start various self-tests that can last between a few minutes and several hours. The program also lets you integrate `smartctl` output as virtual disks and therefore makes it easy to evaluate the collected data.

★★★★★ GSmartControl is a practical graphical interface for `smartctl` and leaves nothing to be desired.



Pam_ihosts 1.1

Function: Access control by Pam module
Source: *https://github.com/ColumPaget/pam_ihosts*
License: GPLv3
Alternatives: Iptables



```
account    required  pam_ihosts.so user=klaus syslog allow-ip=192.168.250.30 all
ow-mac=b8:27:eb:fa:ed:6b

Jul 25 21:19:48 LMLab-1504_a sshd: pam_ihosts ALLOW: user=[klaus] rhost=[tcpdesk
top.vcondt.de] ip=[192.168.250.30] device=[eth0] mac=[00:26:18:a9:39:c6] region=
[(null)]
Jul 25 21:19:48 LMLab-1504_a sshd[11186]: Accepted password for klaus from 192.1
68.250.30 port 52811 ssh2
Jul 25 21:19:48 LMLab-1504_a sshd[11186]: pam_unix(sshd:session): session opened
 for user klaus by (uid=0)
Jul 25 21:19:40 LMLab-1504_a systemd: pam_unix(systemd-user:session): session op
ened for user klaus by (uid=0)
Jul 25 21:19:48 LMLab-1504_a systemd-logind[501]: New session 52 of user klaus.
Jul 25 21:20:03 LMLab-1504_a sshd: pam_ihosts DENY: user=[klaus] rhost=[pib01.vc
ondt.de] ip=[192.168.250.64] device=[eth0] mac=[b8:27:eb:5f:ce:11] region=[(null
)]
Jul 25 21:20:03 LMLab-1504_a sshd[11256]: Failed password for klaus from 192.168
.250.64 port 32896 ssh2
Jul 25 21:20:03 LMLab-1504_a sshd[11256]: fatal: Access denied for user klaus by
 PAM account configuration [preauth]
Jul 25 21:21:37 LMLab-1504_a sshd: pam_ihosts ALLOW: user=[klaus] rhost=[192.168
.250.216] ip=[192.168.250.216] device=[eth0] mac=[b8:27:eb:fa:ed:6b] region=[(nu
ll)]
Jul 25 21:21:37 LMLab-1504_a sshd[11266]: Accepted password for klaus from 192.1
```

The PAM module restricts access to Linux computers. You can decide which hosts are allowed, either via the IP or MAC address or through regional association. However, actual user authentication needs to be working before Pam_ihosts has its say.

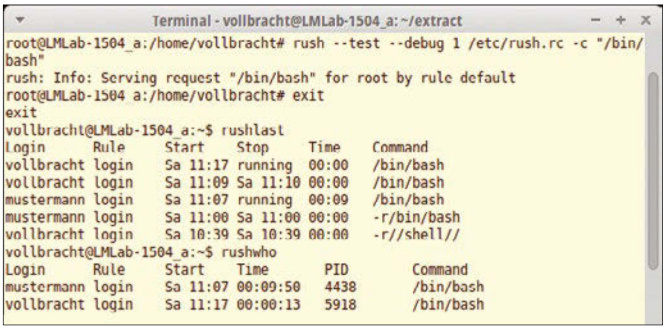
System administrators integrate the module into the computer's PAM configuration and edit the corresponding files in the `/etc/pam.d` directory. They can determine the behavior using various options. For example, there is a username or a list of accounts separated by commas in `user`. An `*` includes all, an `!` allows a reversal, and `allow-ip[s]` and `allow-mac[s]` control access for specific IP or MAC addresses. IPv6 support is currently pending; Pam_ihosts only understands the IPv4 format. If the target machine has several network interfaces, administrators can specify the desired one in `allow-def[s]`.

Admins also can allow or block specific regions. They can retrieve the information from the Regional Internet registries, which are linked from the Pam_ihosts documentation. These text files are in the configuration file at `region-files`; `allow-region[s]` lists one or more allowable ranges.

★★★☆☆ Pam_ihosts is an interesting method for access control. As with all PAM extensions, care is need during the setup. There are deductions for the missing IPv6 support. ■■■

Rush 1.7

Function: Restrict shell access
Source: *http://www.gnu.org/software/rush*
License: GPLv3
Alternatives: Rbash, Rssh



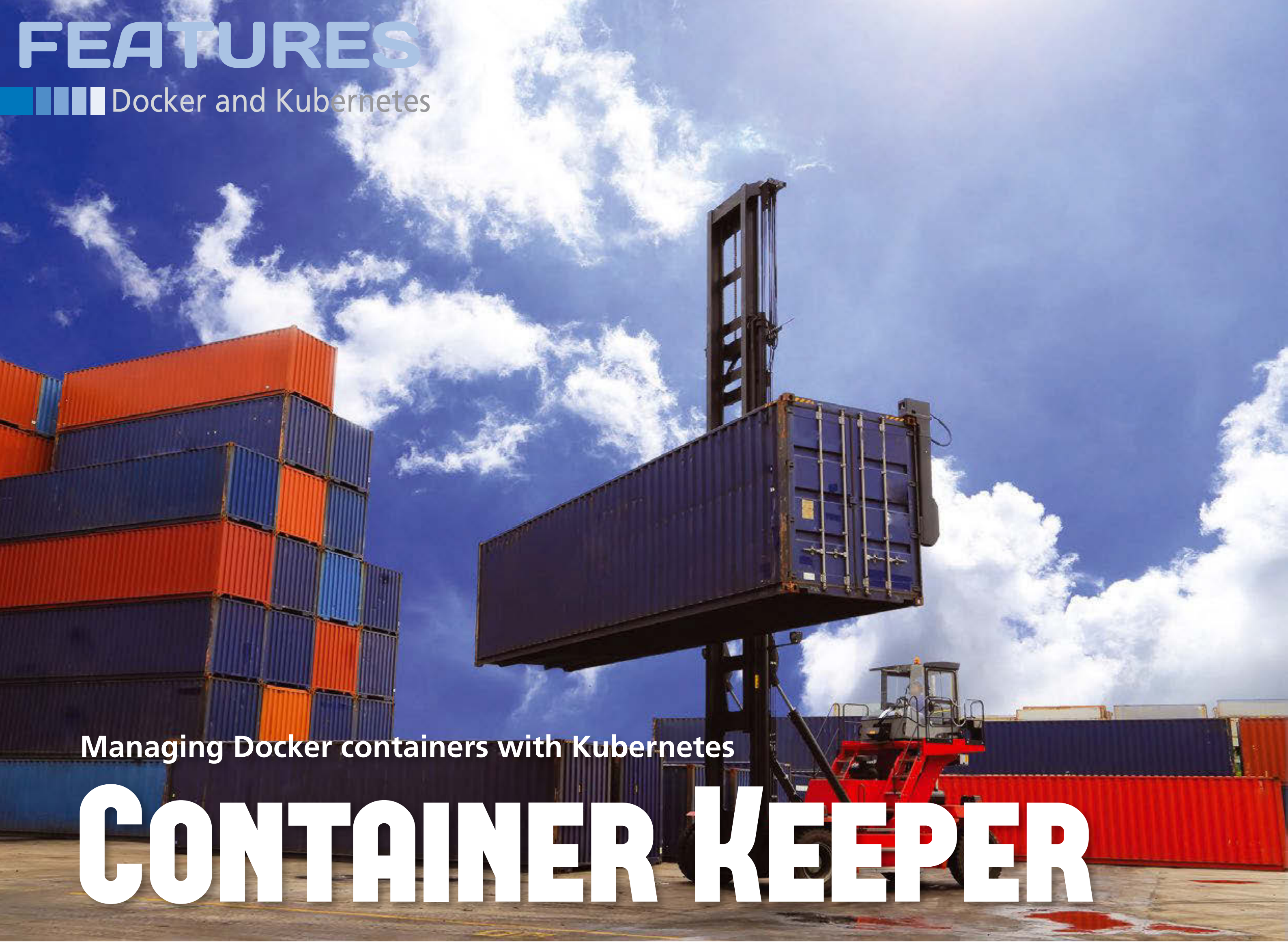
```
root@LMLab-1504_a:/home/vollbracht# rush --test --debug 1 /etc/rush.rc -c "/bin/
bash"
rush: Info: Serving request "/bin/bash" for root by rule default
root@LMLab-1504_a:/home/vollbracht# exit
exit
vollbracht@LMLab-1504_a:~$ rushlast
Login      Rule     Start    Stop     Time    Command
vollbracht login    Sa 11:17 running  00:00   /bin/bash
vollbracht login    Sa 11:09 Sa 11:10 00:00   /bin/bash
mustermann login    Sa 11:07 running  00:09   /bin/bash
mustermann login    Sa 11:00 Sa 11:00 00:00   -r/bin/bash
vollbracht login    Sa 10:39 Sa 10:39 00:00   -r//shell//
vollbracht@LMLab-1504_a:~$ rushwho
Login      Rule     Start    Time        PID        Command
mustermann login    Sa 11:07 00:09:50    4438       /bin/bash
vollbracht login    Sa 11:17 00:00:13    5918       /bin/bash
```

Using the restricted user shell, Linux administrators can define a limited range of functions with which their users can work at the command line. The GNU tool can be used as a login shell and is also well suited for chroot operations. The guide shows examples for accesses via `scp`, `sftp`, `rsync`, and `svn`.

The `/etc/rush.rc` configuration file contains the rules. System administrators can create them in a text editor of their choice. The key word `rule` initiates a new definition. Instructions behind this – such as `command`, `uid`, and `match` – define when a rule is activated. For some conditions, it is sufficient for the administrator to specify a match pattern that can also contain regular expressions. Other options such as `transform`, `chroot`, and `chdir` define the associated actions. For example, they rewrite the original command line, start a chroot environment, or cause a directory change.

Rush executes the first matching rule by default. With such fall-through statements, the tool searches for the next matching rule after executing the previous one. The tool maintains its own WTMP and UTMP files so that administrators can keep an eye on the use of Rush. The `rushwho` and `rushlast` commands evaluate them and write information to standard output.

★★★★☆ Rush provides lots of options for limiting access to the shell. However, creating a workable configuration is pretty time consuming and involves carefully studying the documentation. ■■■



Managing Docker containers with Kubernetes

Container Keeper

After you jump onto the container bandwagon, you will find yourself looking for high-performance solutions for managing the Docker landscape. Several vendors offer special operating system images with built-in container management tools. Red Hat uses Atomic with Google's Kubernetes management tool. By Thorsten Scherf

Running applications in containers is becoming increasingly popular. Containers offer many benefits compared with conventional virtual machines. Docker is a popular container system for Linux that needs only a very minimal base system, so using a conventional, multipurpose operating system with its large collection of miscellaneous components adds a huge overhead if you know all you really want to do is host containers.

The container environment lends itself to portability. In large enterprise environments, IT managers do not want to worry about the type of system a container runs on. The focus is on defining an application with the necessary requirements and deploying it on existing resources. Whether the application actually runs in a container on System A or System B is unimportant.

The need for portability and efficiency has led to development of some special Linux distributions tailored for the container environment. These special distros offer a uniform operating environment, include container management tools, and perhaps most importantly they are optimized for containers – without the feature bloat associated with multipurpose systems.

In addition to today's popular init system, systemd, and some basic kernel components such as SELinux and CGroups, these container systems only use a very small software stack. The toolkit obviously includes Docker as the container engine and, often, Kubernetes [1], a container management and orchestration tool by Google.

The operating system image can operate in many different environments. For example, the image will run on classical bare-metal systems, but it also works for public and private cloud environments, such as Google Cloud Engine (GCE), Red Hat OpenShift, OpenStack, or Amazon Web Services (AWS).

Red Hat's Project Atomic [2] is one of these container-based operating systems. Atomic was specially designed for use in containers on the basis of Docker and Kubernetes. Project Atomic is the upstream project for several other images. For example, Red Hat [3], Fedora [4], and CentOS [5] rely on the Atomic proj-

Lead Image © kamonrat meunklad, 123RF.com

ect to create their own cloud images for use with Docker containers.

These Atomic images are very different from a garden-variety Linux distribution. For example, Atomic does not use a package manager and instead relies on the `rpm-ostree` tool. Atomic updates are possible for the Atomic hosts because the complete operating system instance resides in a single filesystem path below the `/ostree/deploy` folder. At boot time, the latest version of the operating system is then mounted below the root filesystem, where only the `/etc` and `/var` directories are writable. All other folders under `/ostree` are only readable.

An update now simply means copying a complete new operating system instance from the update server to `/ostree/deploy` and applying the changes to the configuration files below `/etc` to the new operating system instance. The `/var` directory is shared between all instances, because shared components such as the user's home directory live there. (The `/home` folder is only a symbolic link to `/var/home`.) To start the new instance of the operating system, reboot the host system.

You cannot install additional software on an Atomic host as you would with RPM or Yum. Instead, you should either run the software in a separate container on the Atomic host or copy to a folder below `/var`. In this case, make sure you have statically compiled the programs, and make all changes in each operating system instance in this folder using the `rpm-ostree` tool – not manually. Using `rpm-ostree update` lets you perform an update of the system. If this update does not work the way you imagined it would, you can restore the system to its original state using `rpm-ostree rollback`. Instead of `rpm-ostree`, you can also simply call the `atomic` tool, which points to `rpm-ostree` through a soft link.

Working with Docker Containers

If you are familiar with Docker containers, you know they are also based on images. These images come either from a central or local Docker registry server or are generated with the help of a Docker file. You can call `docker run` to then start the desired application within a container. The following example shows the famous Hello World in a Fedora container:

```
docker run fedora /bin/echo "hello world"
```

The image with the name `fedora` is not present locally at this time; instead, Docker downloads it independently from the predefined registry server and then runs the `/bin/echo "hello world"` command within the Fedora instance. Then the container terminates. A call to `docker ps -a` displays all containers running on the host.

Instead of using the `echo` command, you could, of course, call a script at this point to launch a preconfigured web server. If the server uses a database back end, create an additional container with just this database and link the two. In small environments, this approach is certainly perfectly okay, but as of a certain size, you would need a solution that scaled better. For example, you would want the ability to start a container or a set of containers on remote hosts. It is also useful to define a status for the applications. If you use Docker to start a container on a host, there are no guarantees that the container will restart on another system in case of a host failure.

Container Orchestration with Kubernetes

Kubernetes offers management and orchestration for container environments. The tool consists of a wide range of services, some of which run on a control system, some on the master host, and others on each Docker host (aka, "the minions"). The app service on the master provides a REST API as a communication channel through which it receives service instructions from clients. An instruction might, for example, generate a specific container on an arbitrary minion host, or a *pod* in Kubernetes-speak.

Usually a pod houses containers for services you would like to install together on conventional systems. A file in JSON format contains all the necessary information. For example, what image should be used for the pod's container and the port on which to listen for services within the container. The minion hosts run an agent service, "kubelet," and receive instructions from the master.

The `etcd` service is used as the communication bus. Etcd is a distributed key/value database [6] that relies on simple HTTP `GET` and `PUT` statements. The etcd database stores the configuration and status information for the Kubernetes cluster and returns the data when needed in JSON format. The `kubelet` service on a minion host constantly queries the database for changes and, if necessary, implements the changes. For example, the database can contain a list of all minion hosts in the cluster. This information is then used by the app service to find hosts on which to generate new containers.

Installing the Atomic Host

To dive into the world of Kubernetes, download one of the Atomic host images available for Red Hat Enterprise [3], Fedora [4], or CentOS [5] and install it in your virtualization environment. For this article, I used a local KVM-based installation on Fedora 21 with a CentOS 7 Docker image. The image is easily installed using the `virt-manager` tool or `virt-install`. For setup instructions for different virtualization environments, see the Red Hat documentation [7]. Note that newer versions of Fedora, and updates for CentOS 7 have appeared since the versions used in this article. Con-

LISTING 1: Meta Configuration Files

```
# mkdir /tmp/atomic/
# cd /tmp/atomic/
# cat > meta-data
instance-id: Atomic0
local-hostname: atomic-00
_eof
# cat > user-data
#cloud-config
password: atomic
chpasswd: {expire: False}
ssh_pwauth: True
ssh_authorized_keys:
   - ssh-rsa AAA...SDvz centos@atomic.example.com
_eof
# genisoimage -output
```

tainer technologies are in rapid development, so you might find some differences from this configuration in your own environment, but the concepts and basic procedures are similar.

The first time you create a virtual machine, you will need to provide a CD in the form of an ISO file. The file contains basic information about the virtual Atomic system, such as the machine name and the password for the default user. You can pass in an SSH key to log on to the system or the desired network configuration. Create the metadata and user data files for this purpose and use them to generate the ISO file (Listing 1); then, provide the file to the Atomic host as a virtual CD drive. When you first start the system, the `cloud-init` service [8] parses the information you provided and configures the system.

LISTING 2: Definition of a Pod

```
{
    "apiVersion": "v1beta1",
    "kind": "Pod",
    "id": "apache-dev",
    "namespace": "default",
    "labels": {
       "name": "apache",
       "stage": "dev"
    },
    "desiredState": {
       "manifest": {
          "version": "v1beta1",
          "id": "apache-dev",
          "volumes": null,
          "containers": [
             {
                "name": "master",
                "image": "fedora/apache",
                "ports": [
                   {
                      "containerPort": 80,
                      "hostPort": 80,
                      "protocol": "TCP"
                   }
                ],
             }
          ],
          "restartPolicy": {
             "always": {}
          }
       },
    },
}
```

If the installation and configuration work, you can then log on to the virtual system to perform an update. In this case, a new instance of the system is downloaded and then activated at the next system start-up time:

```
ssh centos@atomic.example.com
rpm-ostree upgrade
systemctl reboot
```

Because this system is the master host, you can install a second host with the same image directly after configuration. The second host will act as the minion host running the container pods. Of course, at this point, you can install as many minions as you like. A single minion host, however, is enough to understand the basic functionality of Kubernetes. To set up the minion host, generate a second virtual machine and, as described in Listing 1, an additional ISO file, which is then available for the minion host installation. After the installation, update this system and restart.

Once the master and minion are up to date, add the two computers to the `/etc/hosts` file and modify the Kubernetes configuration file `/etc/kubernetes/config`. Enter the master server on both systems via the `KUBE_ETCD_SERVER` variable. The current version of Kubernetes only supports a single master, but this will change in future releases. On the master, modify two more files: the `/etc/kubernetes/apiserver` and `/etc/kubernetes/controller-manager` files. Define the hostname and the port for the API service, as well as the minion server hostname. Following this, start all the necessary services on the master and then make sure everything is working correctly:

```
systemctl start etcd kube-apiserver ↩
  kube-controller-manager kube-scheduler
systemctl enable etcd kube-apiserver ↩
  kube-controller-manager kube-scheduler
systemctl status etcd kube-apiserver ↩
  kube-controller-manager kube-scheduler
```

On the minion host, you additionally need to configure the `/etc/kubernetes/config` file to customize the minion agent's configuration. Open the `/etc/kubernetes/kubelet` file and add the hostname, port, and IP address on which you want the service to listen. Then restart the necessary services:

```
systemctl start kube-proxy kubelet docker
systemctl enable kube-proxy ↩
   kubelet docker
systemctl status kube-proxy ↩
   kubelet docker
```

At this point, you should see the minion host on the master. The `kubectl` tool is used to communicate with the API server:

```
kubectl get minion
NAME
atomic-host-001
```

In the next step, you can create your first pod. As a reminder, this means one or more containers that are provided on one of the available minion hosts. The definition of the pods relies on a file in JSON format, where you define all the information for the pod. This information includes the Docker image to use, port services, and optional port mapping between the container and host. You can also decide on the host filesystems you want to bind to the container. This step is especially important because, if you don't bind the container to a host filesystem, any data that changes within the container is lost after terminating the container.

Each pod can be equipped with one or more labels. For example, you could assign the label `name = apache` and `stage = prod` in the JSON file for all Apache servers

LISTING 3: Kubectl Shows Active Pods

```
kubectl get pods
NAME         IMAGE(S)        HOST               LABELS                     STATUS
apache-dev   fedora/apache   atomic-host-001/   name=apache,stage=dev      Running
```

in production. With a corresponding query via `kubectl`, you can then very easily identify your production Apache servers and discover which minions they are currently running on. But first you need to create your first pod with the file in Listing 2. Call the `kubectl` tool as follows:

```
kubectl create -f /tmp/apache-pod.json
```

In the background, the Docker process starts its work on the minion and begins to download the `fedora/apache` image if it does not already exist. This download can take quite a while. When you call `kubectl` again, you should see that the container is active (Listing 3).

To see if the Apache service in the container is working as usual, make a simple call to `curl`:

```
curl http://atomic-host-001
Apache
```

If you have multiple Apache containers running in your environment, you can restrict the output of `kubectl get pods` based on the previously defined labels. The command

```
kubectl get pods -l name=apache ↩
   -l stage=prod"
```

tells Kubernetes to show you only the containers with two labels: `name=apache` and `prod=stage`.

As you can see in Listing 3, the definition of the pod also contains a note to the effect that a container needs to be immediately restarted in the event of an error (`restartPolicy:always`). Finding out if this works is easy: Log onto the minion host via SSH and tell Docker to display the currently active container (Listing 4).

Now terminate the container manually by entering:

```
docker stop a9548bd9ecb1
```

After a short time, you will notice that docker automatically launches the container again. Watch the value in the *CREATED* column in the output from `docker ps` before and after manually stopping the container.

LISTING 4: Docker Listing Active Containers

```
# docker ps
CONTAINER ID   IMAGE                  COMMAND            CREATED          STATUS ...
a9548bd9ecb1   fedora/apache:latest   "/run-apache.sh"   9 minutes ago    Up 9 minutes ago ...
```

Services and Replication

Kubernetes supports two other very interesting features that I have not mentioned so far. Using a replication controller, you can scale pods horizontally. On the basis of a pod label, you can tell Kubernetes to provide the pod *x* times. Kubernetes then generates the desired number of instances of the container and makes them available on the existing minions. Kubernetes also handles the task of keeping the number of instances up to date. For example, if you stipulated that you want to have four instances of your Apache pods, Kubernetes would automatically start or stop Apache Docker instances until the number of instances corresponds to the definition.

Even if the individual containers get IP addresses from a network block defined previously in Docker, access to the individual instances of a pod are more typically via services. To offer this access, Kubernetes drops a kind of abstraction layer over specific pods of the same type and assigns a single IP address to this service. Access to the individual instances of a pod is then via that service IP. To allow this to happen, the request to the service IP is forwarded to the individual minions, and the kubelet proxy service on the minions is responsible for forwarding the request to the correct container.

You can imagine such a service as a kind of load balancer for a specific set of pods (Figure 1). For this system to work, every minion host must have an additional network segment from which the pod is then assigned an IP address. The kubelet proxy services forward the request to exactly this IP address. However, the only cloud provider that offers such a network configuration out of the box is the Google cloud platform. For all other providers, even if you use the Atomic image from this article in a local virtualization environment, a manual Docker service configuration is required. See the Docker documentation [9].

The `flannel` tool [10], which is included in the most recent releases of the Atomic image, makes it very easy to set up overlay networks, which Docker can then draw on, thus saving you from extensive manual configuration.

Conclusions

Even though Kubernetes is currently profoundly beta and the tool's emphasis on the Google cloud engine is obvious, the gains in flexibility compared with plain vanilla Docker installations are already huge. Kubernetes defines an additional abstraction layer for the existing hardware resources and makes them available as a large pool of hardware.

Which system ultimately runs the container is unimportant. Kubernetes independently pools the necessary resources and starts the container wherever resources are available. Through the use of replication controllers, the framework also helps to scale existing containers horizontally.

Kubernetes services help to facilitate access to a service through a single IP address, thus eliminating the cumbersome process of discovering container IP addresses. This feature is particularly valuable when you consider the fact that containers can be short lived; they are typically restarted just a short time later on a different host; thus, their IP address might change. Services abstract these changes, removing the need for configurations such as upstream load balancers. ■■■

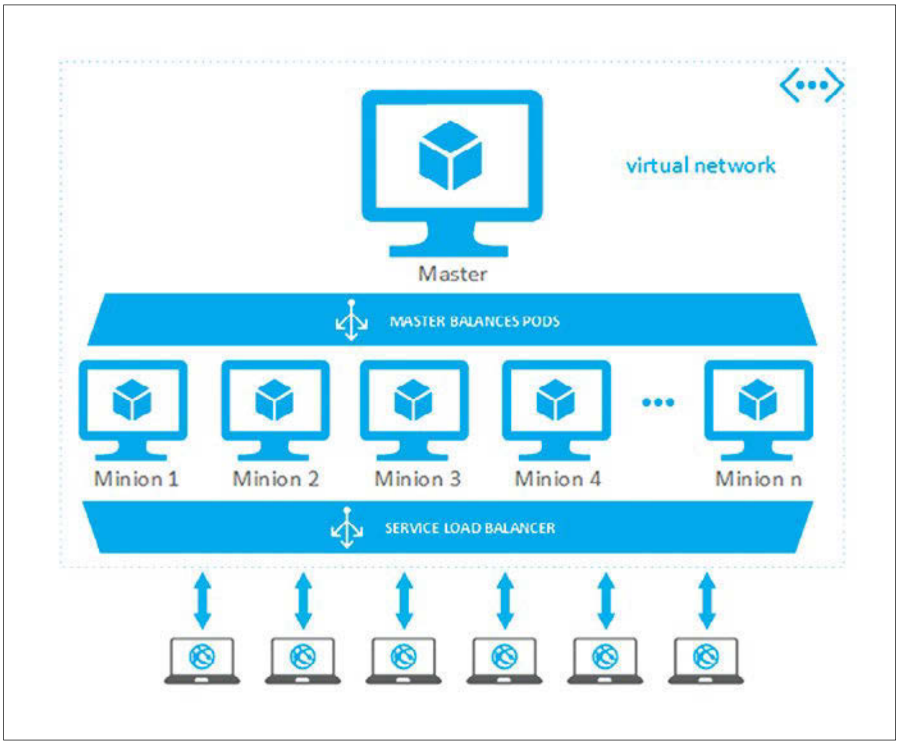


Figure 1: Kubernetes offers a service load balancer that supports access to several pods via a single IP address.

INFO

[1] Google Kubernetes: *https://github.com/kubernetes/kubernetes*

[2] Project Atomic: *http://www.projectatomic.io/*

[3] Red Hat Enterprise Linux 7 Atomic host: *http://www.redhat.com/en/about/blog/small-footprint-big-impact-red-hat-enterprise-linux-7-atomic-host-beta-now-available*

[4] Fedora Atomic host: *https://getfedora.org/en/cloud/download/*

[5] CentOS 7 Atomic host: *http://buildlogs.centos.org/rolling/7/*

[6] Etcd API documentation: *https://github.com/coreos/etcd/blob/master/Documentation/api.md/*

[7] Setup instructions for an Atomic VM *https://access.redhat.com/articles/rhel-atomic-documentation/*

[8] Cloud-init documentation: *https://cloudinit.readthedocs.org/en/latest/*

[9] Docker network documentation: *http://docs.docker.com/engine/userguide/networking/*

[10] Flannel overlay network: *https://github.com/coreos/flannel/*

AUTHOR

Thorsten Scherf is a Principal Consultant for Red Hat EMEA. You can meet him as a speaker at conferences. He is also a keen marathon runner whenever time and family permit.

EXPERT TOUCH

FREE DVD
The world's greatest Live distro!
KNOPPIX 7.4.2
LINUX SHELL HANDBOOK
LINUX SHELL HANDBOOK
6th Edition
LINUX Special
DISCOVER
THE LANGUAGE OF THE LINUX EXPERTS
NEW EDITION! Includes updates for systemd
300+ OF THE BEST BASH COMMANDS
Bash Baccalaureate
Keep this comprehensive guide as a permanent command reference!
Create your own Bash scripts
Configure partitions, permissions, devi
and user accounts
Manage and troubleshoot processes
WWW.LINUX-MAGAZINE.COM

6th Edition!

Linux professionals stay productive at the Bash command line – and you can too!

The Linux Shell special edition provides hands-on, how-to discussions of more than 300 command-line utilities for networking, troubleshooting, configuring, and managing Linux systems. Let this comprehensive reference be your guide for building a deeper understanding of the Linux shell environment.

You'll learn how to:

- Filter and isolate text
- Install software from the command line
- Monitor and manage processes
- Configure devices, disks, filesystems, and user accounts
- Troubleshoot network connections
- Schedule recurring tasks
- Create simple Bash scripts to save time and extend your environment

The best way to stay in touch with your system is through the fast, versatile, and powerful Bash shell. Keep this handy command reference close to your desk, and learn to work like the experts.

FREE DVD INSIDE!

The world's greatest Live distro!

KNOPPIX 7.4.2

ORDER ONLINE:

shop.linuxnewmedia.com/specials

Hashes, salt, and pepper

Salt and Pepper

Cryptographic hash functions help you protect your passwords, but hashing is only secure if properly understood. *By Tobias Eggendorfer*

Hash functions are an integral part of computer science – and not just with databases and checksums. Hashes were originally intended for storing data efficiently in memory, but the hashing concept has evolved into a technique for securely storing passwords.

Linux writes the password hash values to the `/etc/shadow` file, which you can only read if you have root privileges. But even if you have the root password, you'll find it difficult to learn any useful access information. The function used to store the password hash values in `etc/shadow` is a one-way function, which means you can't work backward from the hash value to create the original password – at least in theory. As you'll learn in this article, attackers still sometimes manage to crack these supposedly irreversible hash functions.

What is a Hash?

The idea of a hash is simple: An address is calculated from the value that is to be stored. Suppose, for example, you need to store the four user names *Fritz*, *Laempel*, *Max*, and *Moritz*. A hash function would calculate a numeric value from these names.

The simplest option is to use letters in the alphabet (A = 1, B = 2, etc.), which the program then uses to form a sum of the digits from the letter values in the name (Table 1). The hash value must be in this range because only a limited number of memory locations are available.

The example in Table 1 has seven available memory locations (from 0 to 6). The hash value is therefore determined by the sum modulo 7. Anyone who doesn't fancy computing the numerical values for *Fritz*, *Laempel*, *Max*, and *Moritz* manually can use Bash with the help of Perl, as shown in Listing 1.

Each name first ends up in the `name` variable via the `for` loop; `echo` sends the name through a pipe to `tr`, which replaces all lowercase and uppercase letters and passes the result to a Perl script. The Perl script then splits the character string into individual letters and outputs its ASCII value separated by the `- 64 +` character string. This gives A the value 1, B the value 2, and so on. Anyone who wants to see this happening live should enter `set -x`. To terminate the function again use `set +x` (Figure 1).

In this example, Max and Moritz have the same hash value. If you imagine memory as an array, you can only accommodate one entry in each slot and

TABLE 1: Simple Hash Function

Value	Sum	Hash
Fritz	79	2
Laempel	64	1
Max	38	3
Moritz	101	3

LISTING 1: Bash Hash

```
$ for name in "Fritz" "Laempel" "Max" "Moritz"; do
  echo -n $name": "; echo -n $name | tr "[:lower:]" "[:upper:]" |
  expr \( `perl -nwle"print join ' - 64 + ', unpack 'C*', $_"` - 64 \) % 7;
  done
Fritz: 2
Laempel: 1
Max: 3
Moritz: 3
```

Lead Image © topnat, 123RF.com

will be unable to handle the task at hand. Theoretical computer science therefore doesn't consider the hash value to be a physical slot, but rather a number of a bucket that can accommodate several entries.

With open hashing (Figure 2), Max and Moritz could live in one bucket because each bucket can contain any number of entries. However, the number of entries is limited with closed hashing.

An array of pointers, which can (in principle) contain an unlimited number of elements, is a suitable data structure for open hashing. At each index of the array a linked list to the elements of this bucket is kept. Open hashing is always useful if the size of the array is comparable to the number of elements, *n*, for all the values you want to store, because then, on average, only one entry per index is stored, allowing for quick access.

The hash function requires a constant time O(1) to calculate the hash value. However, the time it takes to wade through the list in each bucket increases as a linear function of the number of elements. The worst case scenario is that all *n* elements end up in one bucket and the required access time is now O(*n*). In that case, this method offers no advantage over a singly linked list.

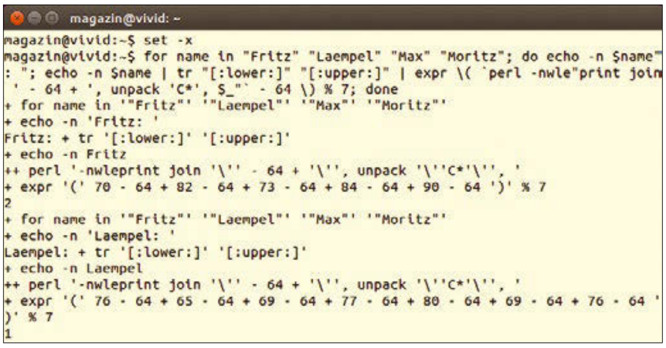
The hash function will ideally distribute the number of values equally across all *m* indices [O(*n*/*m*)], which makes finding elements faster than with a singly linked list. Such a structure is particularly useful if you're looking for a specific value from a large amount of data. Closed hashing limits the number of entries in a bucket and thereby prevents long chains.

Keep a Lid on It

Many real implementations draw the line for the maximum number of entries per bucket at 1; however, higher values are also theoretically possible. The simplest implementation is a two-dimensional array with *n* number of rows (= buckets) and *m* number of columns for the maximum number of elements per bucket. However, this solution isn't ideal for memory usage.

Collisions are now no longer irrelevant; you still have to check whether the bucket has room for another value.

Some methods provide a new hashing (called rehashing) with another hash function as the solution. This rehashing is repeated until no more collisions occur. The worst case scenario is that you'll need a lot of new hash functions, which give rise to different results to find a free space.



```
magazin@vivid:~$ set -x
magazin@vivid:~$ for name in "Fritz" "Laempel" "Max" "Moritz"; do echo -n $name"
: "; echo -n $name | tr "[:lower:]" "[:upper:]" | expr \( `perl -nwle"print join
 ' - 64 + ', unpack 'C*', $_"` - 64 \) % 7; done
+ for name in '"Fritz"' '"Laempel"' '"Max"' '"Moritz"'
+ echo -n 'Fritz: '
Fritz: + tr '[:lower:]' '[:upper:]'
+ echo -n Fritz
++ perl '-nwleprint join '\'' - 64 + '\'', unpack '\''C*'\'', '
+ expr '(' 70 - 64 + 82 - 64 + 73 - 64 + 84 - 64 + 90 - 64 ')' % 7
2
+ for name in '"Fritz"' '"Laempel"' '"Max"' '"Moritz"'
+ echo -n 'Laempel: '
Laempel: + tr '[:lower:]' '[:upper:]'
+ echo -n Laempel
++ perl '-nwleprint join '\'' - 64 + '\'', unpack '\''C*'\'', '
+ expr '(' 76 - 64 + 65 - 64 + 69 - 64 + 77 - 64 + 80 - 64 + 69 - 64 + 76 - 64 '
)' % 7
1
```

Figure 1: Entering set -x in front of the little script in Listing 1 ensures that Bash displays each line before it is executed.

The simplest hash functions always add 1 to the previous hash value; *h*[0] (*h* subscript 0) is the original function, *n* is the number of buckets, and *i* is the *i*th rehashing attempt:

$$h_i = (h_0 + i) \bmod n$$

This process is called linear probing and is a bit like a visit to an Oktoberfest tent: People go from table to table until they find an empty seat. This process causes block formations. In places where everything is already occupied, the next table is likely to be occupied too.

Therefore refinements alternate between looking above and below. Square probing using *i*^2 (*i* squared) instead of *i* is a possible alternative. Collisions are equally likely with the first hash, but the likelihood decreases with rehashing.

Rehashing leads to another problem: Anyone looking for an entry in memory later (Max for example) just needs to apply the function to the name and will then receive the memory address. However, if nothing is stored at this address, you need to keep rehashing until you find Max or an empty or only partially filled bucket, which would mean that Max wasn't saved.

If an entry has been deleted, you need to set a marker at this position so the search is terminated prematurely. The memory space should be less than 80 percent full to keep the probability of collisions low enough.

Adding Security

Cryptography places higher demands on hash functions. A hash function is considered secure if it isn't possible to recreate the output value from the hash value – at least not in an acceptable time. In other words, the function must not be reversible. The probability of finding a second output value for the given hash and the probability that any two output values lead to the same hash value should be very low.

The last two stipulations sound similar, but they differ in a way that is best illustrated with the birthday paradox: Suppose Max is at a party and asks the other guests if anyone's birthday is on the same day as his. An affirmative response is relatively unlikely. If, on the other hand, he wanted to know if any two arbitrary partygoers' birthdays are on the same day, the probability is more than 50 percent, even with as few as 23 people at the party. For the first scenario, 183 people would be needed to get such

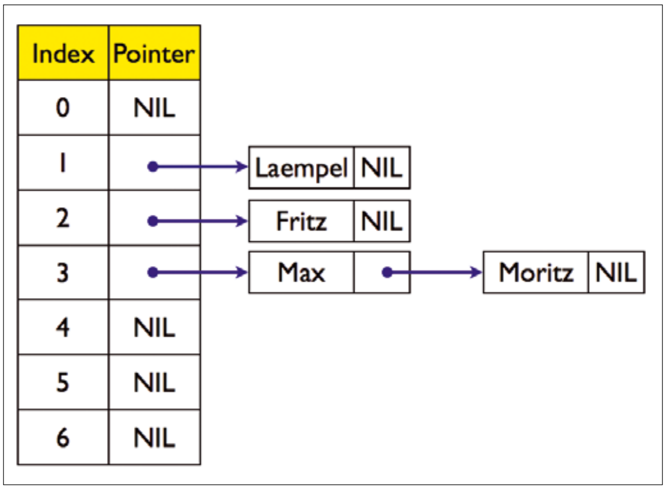


Figure 2: What open hashing looks like using the values in Table 1.

a value. So, from a cryptographic point of view, birthdays are a bad hash value, because a very large number of people (7 billion) map to a very small range of values (365 days).

The range of values for a cryptographically secure hash function is a fundamental part of the design. MD5, for example, uses 128 bits (2^128, i.e., about 3 x 10^38 different values). SHA-1 uses 160 bits (10^48 different values), SHA-256 as many as 256 bits (10^77). A function is considered to be collision-resistant if the probability of a collision is less than 2^(*n*/2) with a hash value of *n* bits.

Like all hash processes, the MD5 algorithm first converts a message to the appropriate length. An integer multiple of 512 bits is required. Padding works in such a way that 1 is appended to the message as the first bit, followed by zeros, until 64 bits remain free at the end. The message length is therefore divisible by 512. A 64-bit value appended to the end (in binary-coded form) specifies how long the original message was [1].

The algorithm then splits the message up into 512-bit blocks and splits these blocks, in turn, into 32-bit blocks. The hash values are then calculated for these blocks, also known as words. Four words are always linked to each other logically four times in different ways, moved bit-by-bit or added. The result is four 32-bit words: A, B, C, and D. The algorithm adds them to the previously calculated values A, B, C, and D. This combination of A, B, C, and D after each round is called a *status vector*.

This continues until the entire message is processed. The MD5 value is then A, B, C, and D in a row. A, B, C, and D receive a fixed output value – the initialization vector – before the calculation.

Cracked

MD5 has been cracked: In 2005, Wang and Yu [2] presented a process in which as few as two appropriately designed 128-bit blocks in the middle of a message are sufficient to make the status vector the same again after the two blocks [3]. Two files created this way may then differ in these 128 bits, but they have the same MD5 hash.

Although 128 bits doesn't sound like a lot, it is enough to prepare an attack. If attackers know which program their victim wants to use, they can incorporate a piece of code that checks which 128-bit block is in the middle. In theory, the query looks like this:

```
if (128-bit block == good version) ↩
  then do_something_good() ↩
  else do_something_evil();
```

The only thing left to do now is to calculate the 128-bit block correctly. Tools such as Evilize [4] are of help here. Anyone who takes a look at the `hello_erase.c` sample program on the project page will recognize the structure above. The selection of the desired function is defined with `goodevil.c`. If the memory block has the content for the `good` process, then the "good" program version should start, otherwise the "evil" one will. The file `crib.h` defines the memory area using a character string that the Evilize program later replaces with two appropriate 128-bit blocks, which lead to the same MD5 hash.

Anyone wanting to try this technique out themselves should extract the Evilize archive and compile the `hello_erase.c` source file:

```
gcc hello-erase.c goodevil.o ↩
  -o hello-erase
```

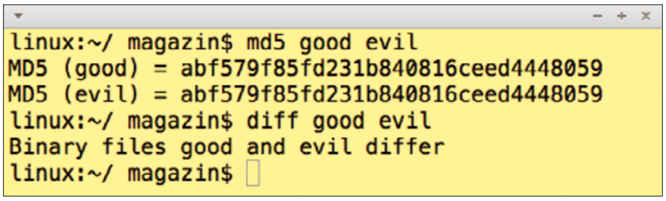
The following call then generates two programs, `good` and `evil`, with the same MD5 hash:

```
./evilize hello-erase -g good -e evil
```

A check using `diff` and `md5sum` shows that the two files are different, but that the hash values are the same (Figure 3).

Passwords

An attack pattern similar to the one described in the previous section is conceivable for cryptographic signatures provided with PostScript, PDF, Word documents, TIFF images, and Flash movies [5]. At the Chaos Communication Congress in 2008, some participants demonstrated how easy it is to forge certificates and signatures using this technology [6].



```
linux:~/ magazin$ md5 good evil
MD5 (good) = abf579f85fd231b840816ceed4448059
MD5 (evil) = abf579f85fd231b840816ceed4448059
linux:~/ magazin$ diff good evil
Binary files good and evil differ
linux:~/ magazin$
```

Figure 3: Two different files with the same MD5 hash value are created by cleverly manipulating 128 bits.

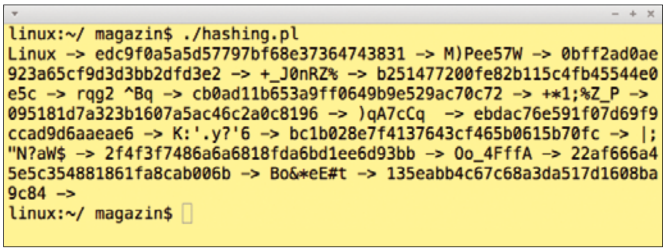
If it's that easy to have the same MD5 hash for two different files, are passwords on a Linux computer really still safe? It depends on how easy it is to find a second password that has the same hash as another. Fortunately, that's not particularly likely because most passwords are likely to have fewer than 16 characters (128 bits), which is the minimum number required for an attack.

Moreover, the fairly simple MD5 algorithm takes up quite a bit of computing time. Even if an attacker calculates the MD5 hash in advance from all the possible passwords, it wouldn't be very practical. With eight-digit passwords consisting of uppercase and lowercase letters and numbers (no punctuation and special characters; i.e., passwords 8 bytes long), there are as many as 6^28 (2 x 10^14) possible combinations; such a list would require about 5 petabytes of memory space.

The approach is therefore not very practical and is only worthwhile for cracking multiple passwords. Brute force attacks are more efficient for one-off jobs.

Behind the Rainbow

One way to reduce memory requirements for a pure MD5 list and also to shorten computing time is to use rainbow tables. The rainbow table method uses a reduction function to generate a new password from a hash. Listing 2 shows an example implemented in Perl. The `reduce` function focuses on the printable ASCII characters ranging between 32 and 127. `md5_hex` calculates a hash value from the password obtained from this reduction function; the reduction function then generates a new password, and so on. Depending on the implementation, a chain of passwords and their hashes is created, and it is possible to restore the original password at any point. Figure 4 shows the Perl script output with *Linux* as the starting password.



```
linux:~/ magazin$ ./hashing.pl
Linux -> edc9f0a5a5d57797bf68e37364743831 -> M)Pee57W -> 0bff2ad0ae
923a65cf9d3d3bb2dfd3e2 -> +_J0nRZ% -> b251477200fe82b115c4fb45544e0
e5c -> rqg2 ^Bq -> cb0ad11b653a9ff0649b9e529ac70c72 -> +*1;%Z_P ->
095181d7a323b1607a5ac46c2a0c8196 -> )qA7cCq  -> ebdac76e591f07d69f9
ccad9d6aaeae6 -> K:'.y?'6 -> bc1b028e7f4137643cf465b0615b70fc -> |;
"N?aW$ -> 2f4f3f7486a6a6818fda6bd1ee6d93bb -> Oo_4FffA -> 22af666a4
5e5c354881861fa8cab006b -> Bo&*eE#t -> 135eabb4c67c68a3da517d1608ba
9c84 ->
linux:~/ magazin$ 
```

Figure 4: The Perl script in Listing 2 generates a rainbow table with a chain of passwords and their hashes from the password *Linux*.

LISTING 2: Perl Script hashing.pl

```
#!/usr/bin/perl -w
use Digest::MD5 qw(md5 md5_hex md5_base64);
sub reduce
{ my $what = shift;
  for ($res='', $i=0; $i<8; $i++)
    { $res.=chr(hex(substr($what,$i*2,2)) % 96 + 32)
    }
  return $res;
}

$password = "Linux";

for ($count=0; $count < 10; $count++)
  {
    print $password." -> ";
    $hash = md5_hex($password);
    $password = reduce($hash);
    print $hash." -> ";
  }
print "\n";
```

This chain may be of any length; but it's not a good idea for it to be too long, because a long chain increases search time and memory requirements. Common implementations reduce memory hunger by storing just the first and the last password from a chain. Otherwise, the memory requirements would be just as big as with a full list of passwords and their hashes.

The process has two catches: First, there is no guarantee you will actually get hold of all the possible passwords, and second, it could lead to collisions and to the same password eventually appearing in several lists. From that point on, two tables would be identical, which massively reduces the efficiency and wastes memory.

In a paper published at the DFRWS conference in Montreal in 2009, Thing and Ying [7] presented a modification of the rainbow tables attack that reduces the risk of collision and improves speed.

Attackers wanting to calculate a password from a hash apply the reduction function to the hash value. They then receive a chain of hashes and passwords to compare with the rainbow tables. If a row ends in a password generated this way, it is clear that the sought-after hash is also in the row. The attacker then constructs the row, again starting with the first password, and compares each hash with the sought-after one. Once the values match, the password is to the left of the hash – and the password is cracked.

It is only worth using rainbow tables if attackers want to crack passwords repeatedly. Brute force attacks are more efficient for one-off attacks. Several ready-made rainbow tables [8] [9] that potential intruders can use are on the Internet. The question is how far they'll get with them. If the sought-after password doesn't appear in the list, a rainbow table attack won't find it either. It works particularly badly with pretty long passwords because most tables assume a password length between six and 10 characters.

Modern Linux systems use a salt, which is appended to the password, to make such attacks more difficult. The salt may be in the `/etc/shadow` file and is therefore not a secret, but the rainbow tables technique fails because a separate table would be needed for every conceivable salt. Instead of a single table, 256 are needed with a salt that's 8 bits long. Linux uses a 12-bit salt; the number of necessary tables thus increases by a factor of 4,096.

Linux uses a separate salt for each password. An attacker therefore needs to design a separate rainbow table for each password-salt combination. It's worth discussing whether to use an individual or a common salt (called pepper).

Pepper is beneficial if, for example, a PHP script writes password hashes to a database. The pepper can then be in the script and the hashes in the database. An attacker reading them via SQL injection will not know the pepper yet, meaning the effort required to create rainbow tables increases significantly.

Conclusions

Hashing can organize data efficiently in memory because an access time of O(1) will, ideally, be possible. Hash functions are suitable for protecting passwords as soon as cryptographically secure hash processes with high collision resistance are used. However, an attacker can try to determine the original password from a hash value using rainbow tables. It only becomes an unrealistic amount of overhead when a salt is involved. ■■■

INFO

[1] RFC to MD5: *https://tools.ietf.org/html/rfc1321*

[2] How to Break MD5 and Other Hash Functions: *http://merlot.usc.edu/csac-f06/papers/Wang05a.pdf*

[3] Collisions for Hash Functions: *http://eprint.iacr.org/2004/199.pdf*

[4] Evilize: *http://www.mathstat.dal.ca/~selinger/md5collision/downloads/evilize-0.2.tar.gz*

[5] A Note on the Practical Value of Single Hash Collisions for Special File Formats: *http://csrc.nist.gov/groups/ST/hash/documents/Illies_NIST_05.pdf*

[6] Creating a rogue CA certificate: *http://www.win.tue.nl/hashclash/rogue-ca*

[7] A Novel Time-Memory Trade-off Method for Password Recovery: *http://dfrws.org/2009/proceedings/p114-thing.pdf*

[8] Free rainbow tables: *https://www.freerainbowtables.com/en/tables2*

[9] Free XP rainbow tables: *http://ophcrack.sourceforge.net/tables.php*

AUTHOR

Tobias Eggendorfer is a professor of IT security in Weingarten (Baden-Wuerttemberg, Germany). He also holds foundation lectures on theoretical computer science. He is also a freelance IT consultant and external data protection officer: *http://www.eggendorfer.info*.

FREE CD
GParted
Manage your partitions with this easy Live CD!
Rescue Systems
ADMIN
Network & Security
FREE CD INSIDE!
DEC/JAN 2016
Rescue Systems
Grml, Rescatux, Knoppix, SystemRescueCd
Which one will save your bacon?
What's new in:
PowerShell 5
PostgreSQL 9.4
Mailman 3.0
Docker Dispatch
• Install and run a Docker environment
• Lock down container security
• Magnum: OpenStack API for containers
EMET
Enhanced Mitigation Experience Toolkit plugs security holes on Windows computers
Configuration Management
Synchronize configuration across multiple hosts
Apache Storm
Analyzing big data
Protocol Analysis
Analysis tools in virtual environments
OpenSMTPD
Highly secure mail server configuration
Security Monitoring
Integrate a SIEM system into your enterprise
Windows • Linux • Unix • Solaris
WWW.ADMIN-MAGAZINE.COM

Harden your systems with Lynis

The Tester

The Lynis testing tool looks for potential security problems and even suggests possibly remedies. *By Tim Schürmann*

To safeguard your system from attack, you'll need to check many components and configuration files for vulnerabilities. This task is worthy of Sisyphus, but never fear – a small tool named Lynis can help you roll that rock. In addition to identifying problems, Lynis offers tips for how to resolve them.

When launched, Lynis [1] performs several hundred individual tests. In each test, the software checks the security of many components. Lynis takes a close look at the configuration files of the installed programs, checks the firewall rules, discovers expired SSL certificates, reports user accounts without a password, and more. According to the company behind Lynis, CISOfy, the tool follows generally accepted security guidelines and standards.

At the end of these tests, Lynis outputs a test report in which it points to the problems it has identified and gives the administrator tips on how to harden the system more effectively. Lynis thus identifies security problems, but it cannot resolve them autonomously; the interpretation of the results is left to the administrator. CISOfy sees the main applications for the tool as security audits, vulnerability scanning, and the first step toward system hardening.

You can launch Lynis directly; there is no need to install. Administrators can thus easily add it to a collection of tools on a rescue USB stick. Lynis also supports plugins to extend the feature scope. In addition to Linux, Lynis runs on other Unix-style systems, including OS X.

Choosing a License

Lynis is available under the GPLv3 and can thus be used without charge in the enterprise. CISOfy also offers a commercial version called Lynis Enterprise, which extends Lynis to include additional features and tools. The tools include a Lynis Collector component, which collects the test results from several computers and feeds the results to a central management console. Lynis Enterprise delivers more comprehensive reports. Among other things, administrators receive an assessment of the computers that are particularly endangered. Finally, CISOfy offers support – but not for the free variant. Lynis Enterprise is available under a subscription model with several levels. The simplest variant costs $1.50 per month and per system. If you need the full feature scope, you can expect to pay $3 per system per month. For more details on Lynis Enterprise, check out the website [2].

Installation

Many Linux distributions have the free Lynis version in their repositories – typically in the *lynis* package. In most cases, the repository will have an older version of the tool. For example, the package manager in Ubuntu 14.10 still offers version 1.5.5, although the latest version when this article was written was Lynis 1.6.4. Because newer versions may be able to discover additional issues, administrators will want always to use the latest version from the Lynis homepage. If you are thinking of using the tool in the long term, you need to keep it up to date yourself.

Once you have the `.tar.gz` archive with Lynis on your hard disk, it makes sense to validate the download by checking the SHA1 or SHA256 checksum. To do so on Linux, for example, type:

```
sha256sum lynis-version.tar.gz
```

Now compare the generated hash with the values that CISOfy provides in the

Lead Image © Leo Blanchette, 123RF.com

File Integrity Information box on the download page [3]. You can only be sure that the archive has not been manipulated if the checksums match. If you want to be double sure, you can also download the digital signature, which is also available from the *File Integrity Information* box. You can then verify the source using GnuPG:

```
wget https://cisofy.com/files/➘
   cisofy-software.pub
gpg --import cisofy-software.pub
gpg --list-keys --fingerprint
```

Instead of `wget`, users on Mac OS X can run `curl`:

```
curl https://cisofy.com/files/➘
   cisofy-software.pub ➘
   -o cisofy-software.pub
```

The fingerprint for CISOfy output with the last command should now be identical to the one returned by the following command:

```
gpg --verify lynis-1.6.4.tar.gz.asc ➘
   lynis-1.6.4.tar.gz
```

You might need to change the version numbers. Also, the fingerprint must match the one printed in the official documentation [4].

Checking Privileges Before Starting

If the checksum and the fingerprint are good, you can finally start up Lynis. To do so, simply unpack the archive and launch the `lynis` script with the `-c` parameter:

```
./lynis -c
```

The `-c` parameter tells Lynis to perform a full set of tests. Without it, Lynis would simply display the help. The command

```
./lynis --view-manpage
```

lets you view the fairly sparse man page. To check whether you have the latest version of Lynis, you can run:

```
./lynis --check-update
```

To inspect all the nooks and crannies of your system, Lynis needs root or administrative privileges. If you launch it as a normal user, the tool might not find all the problems. In any case, Lynis needs write privileges for the directories `/tmp` and `/var/log`. (Test reports will land in the `/var/log` directory.)

After launching, Lynis states the privileges with which it is running, whether or not it can perform all the tests, and whether it can write a logfile below `/var/log` (Figure 1). If you agree with all the settings, you can start the test run by pressing Enter.

Under certain circumstances, Lynis will complain about not having the right file permissions or ownership. You need to remedy this with the commands shown by Lynis; only then can you run the tool. On Linux, the following command will remedy all the ownership problems Lynis complains about with one action:

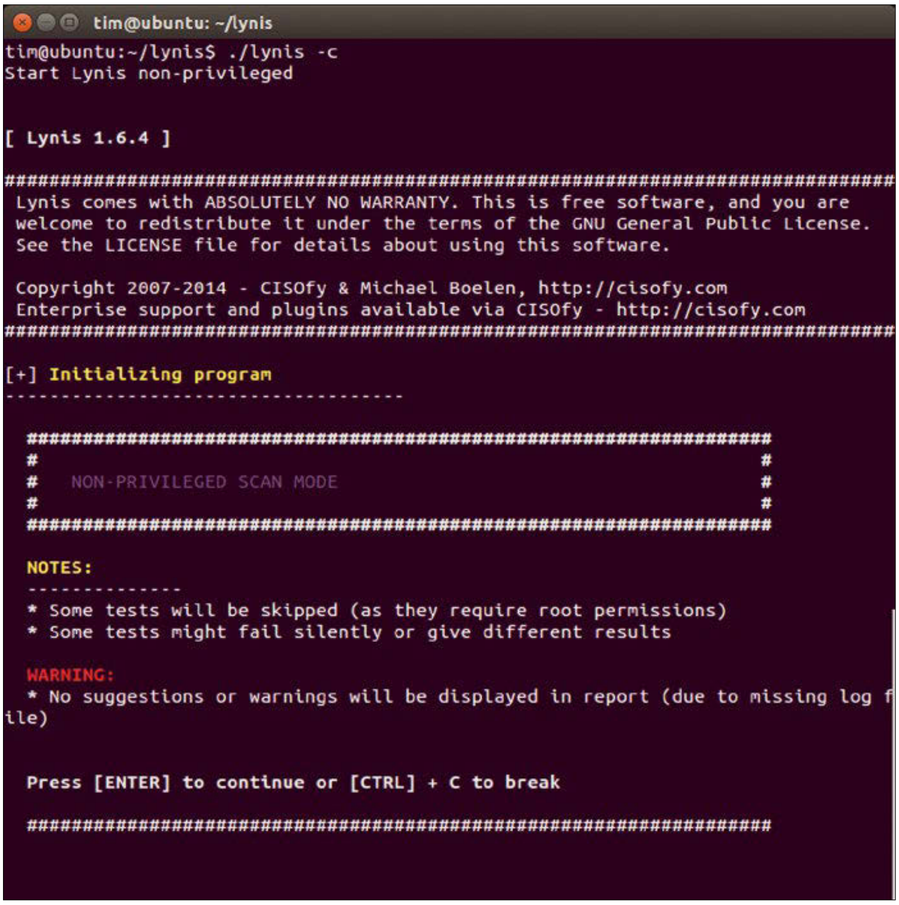
```
sudo chown root:root ./include/*
```

Once Lynis agrees with the ownership, it again summarizes the scenario. Among other things, Lynis states its program version, the operating system, and the storage location of the logfile and report file. If the logfile and report file end up in the black hole of `/dev/null`, you can assume that Lynis is unable to write to the `/var/log` directory. Currently, there is no option for defining a different storage location. Users can only suppress the logfile by stipulating the `--no-log` parameter.

Starting Your Security Scan

After confirming by pressing the Enter key, Lynis starts its tests and writes the results to your standard output. Results marked in green are the successfully completed tests; red warnings point to problems or vulnerabilities (Figure 2). Administrators simply need to make a note of gray result messages. They show that, for instance, the service is not installed on the system. Messages highlighted in yellow do not point to critical problems, but Lynis does see some scope for improvement. The tool might even suggest some remedies, which appear right at the end of the test report (Figure 3).

After each test, Lynis stops, thus giving the user the opportunity to inspect

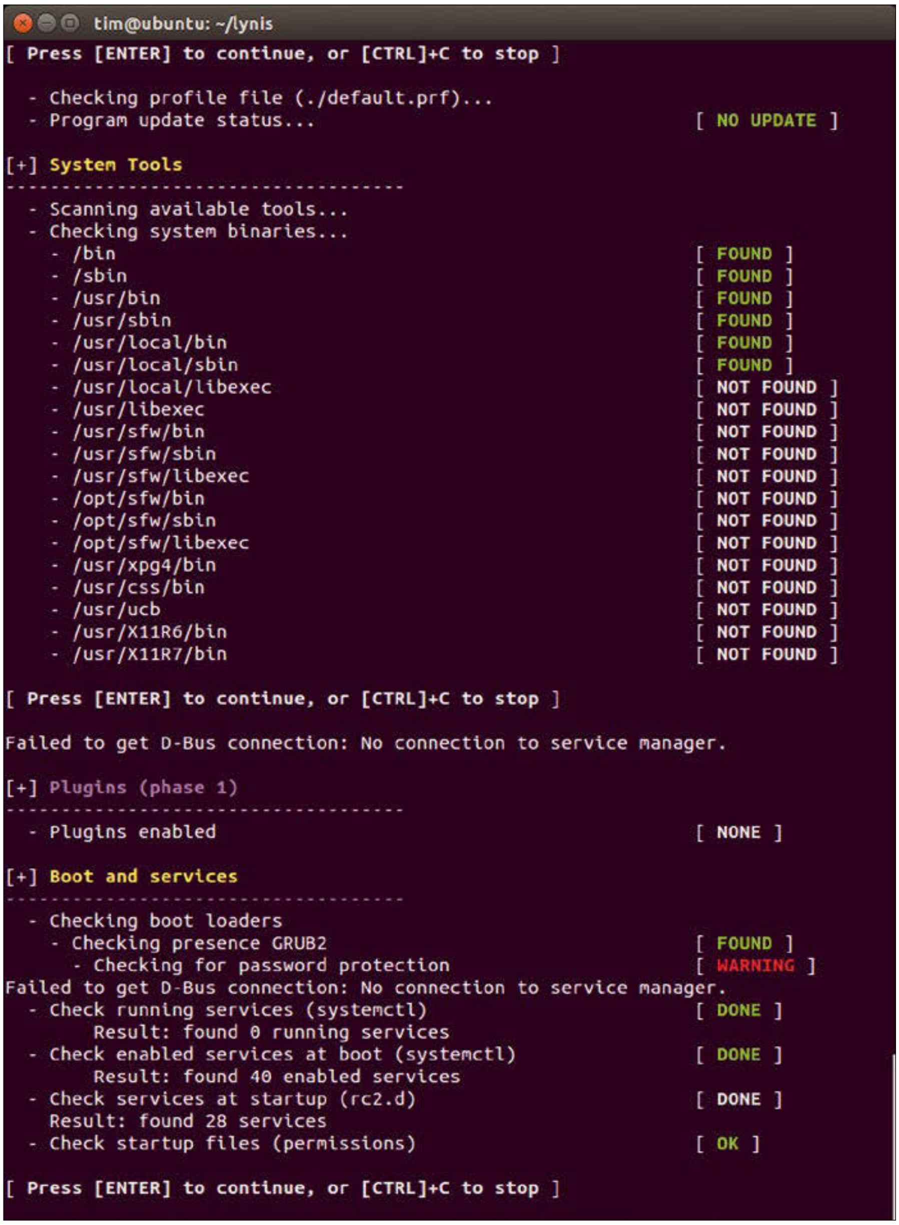


```
tim@ubuntu:~/lynis$ ./lynis -c
Start Lynis non-privileged


[ Lynis 1.6.4 ]

################################################################################
 Lynis comes with ABSOLUTELY NO WARRANTY. This is free software, and you are
 welcome to redistribute it under the terms of the GNU General Public License.
 See the LICENSE file for details about using this software.

 Copyright 2007-2014 - CISOfy & Michael Boelen, http://cisofy.com
 Enterprise support and plugins available via CISOfy - http://cisofy.com
################################################################################

[+] Initializing program
------------------------------------

  ###################################################################
  #                                                                 #
  #   NON-PRIVILEGED SCAN MODE                                      #
  #                                                                 #
  ###################################################################

  NOTES:
  --------------
  * Some tests will be skipped (as they require root permissions)
  * Some tests might fail silently or give different results

  WARNING:
  * No suggestions or warnings will be displayed in report (due to missing log f
ile)


  Press [ENTER] to continue or [CTRL] + C to break

  ###################################################################
```

Figure 1: When a normal user launches Lynis, the tool cannot perform some tests and cannot write a log file below `/var/log`.

the output. The test continues after you press the Enter key. If you want Lynis to simply perform all the tests in succession, just add the `--quick` or `-Q` parameter to the command line. Depending on your terminal, the output from Lynis may use problematic color schemes, or it might be difficult to read. You can stipulate the additional `--reverse-colors` parameter to tell Lynis to adapt the colors for a light background. If this doesn't help, you can also completely disable colors by stipulating the `--no-colors` parameter.

Lynis consists of a collection of shell scripts, each of which checks precisely one system component. For their work, the scripts harness the power of typical system commands. For example, one of the scripts tests the firewall rules by running iptables. You might find that programs cannot be executed – either because of a program error or because the required access privileges for executing the program are missing. Lynis then deems the test to have failed and issues a warning. You can see an example of this behavior in Figure 2, where access to D-Bus failed. Administrators thus always need to check whether Lynis really has found a problem or whether the test was canceled. It makes sense to check out the error message that appears below the test.



```
tim@ubuntu: ~/lynis
[ Press [ENTER] to continue, or [CTRL]+C to stop ]

  - Checking profile file (./default.prf)...
  - Program update status...                                  [ NO UPDATE ]

[+] System Tools
------------------------------------
  - Scanning available tools...
  - Checking system binaries...
    - /bin                                                    [ FOUND ]
    - /sbin                                                   [ FOUND ]
    - /usr/bin                                                [ FOUND ]
    - /usr/sbin                                               [ FOUND ]
    - /usr/local/bin                                          [ FOUND ]
    - /usr/local/sbin                                         [ FOUND ]
    - /usr/local/libexec                                      [ NOT FOUND ]
    - /usr/libexec                                            [ NOT FOUND ]
    - /usr/sfw/bin                                            [ NOT FOUND ]
    - /usr/sfw/sbin                                           [ NOT FOUND ]
    - /usr/sfw/libexec                                        [ NOT FOUND ]
    - /opt/sfw/bin                                            [ NOT FOUND ]
    - /opt/sfw/sbin                                           [ NOT FOUND ]
    - /opt/sfw/libexec                                        [ NOT FOUND ]
    - /usr/xpg4/bin                                           [ NOT FOUND ]
    - /usr/css/bin                                            [ NOT FOUND ]
    - /usr/ucb                                                [ NOT FOUND ]
    - /usr/X11R6/bin                                          [ NOT FOUND ]
    - /usr/X11R7/bin                                          [ NOT FOUND ]

[ Press [ENTER] to continue, or [CTRL]+C to stop ]

Failed to get D-Bus connection: No connection to service manager.

[+] Plugins (phase 1)
------------------------------------
  - Plugins enabled                                           [ NONE ]

[+] Boot and services
------------------------------------
  - Checking boot loaders
    - Checking presence GRUB2                                 [ FOUND ]
      - Checking for password protection                      [ WARNING ]
Failed to get D-Bus connection: No connection to service manager.
  - Check running services (systemctl)                        [ DONE ]
      Result: found 0 running services
  - Check enabled services at boot (systemctl)                [ DONE ]
      Result: found 40 enabled services
  - Check services at startup (rc2.d)                         [ DONE ]
    Result: found 28 services
  - Check startup files (permissions)                         [ OK ]

[ Press [ENTER] to continue, or [CTRL]+C to stop ]
```

Figure 2: Test results marked in yellow or red definitely need to be checked by an administrator. In this case, Lynis is complaining about problems with password security.

LISTING 1: Integrating Lynis with Cron

```
#!/bin/sh
DATE=$(date +%Y%m%d)
# Launch Lynis:
/usr/local/lynis/lynis -c --cronjob > /var/log/lynis-output.${DATE}.txt
# Back up generated log files:
mv /var/log/lynis.log /var/log/lynis.${DATE}.log
mv /var/log/lynis-report.dat /var/log/lynis-report.${DATE}.dat
```

When finished, Lynis delivers some statistics, as shown in Figure 4. You will find a somewhat more exhaustive test report in the `/var/log/lynis.log` file. Its counterpart, `/var/log/lynis-report.dat`, contains a summary of the test run and the improvements suggested by Lynis. But caution is advisable: as soon as you launch Lynis the next time, it will overwrite the existing logfile and report file.

Automated Scans

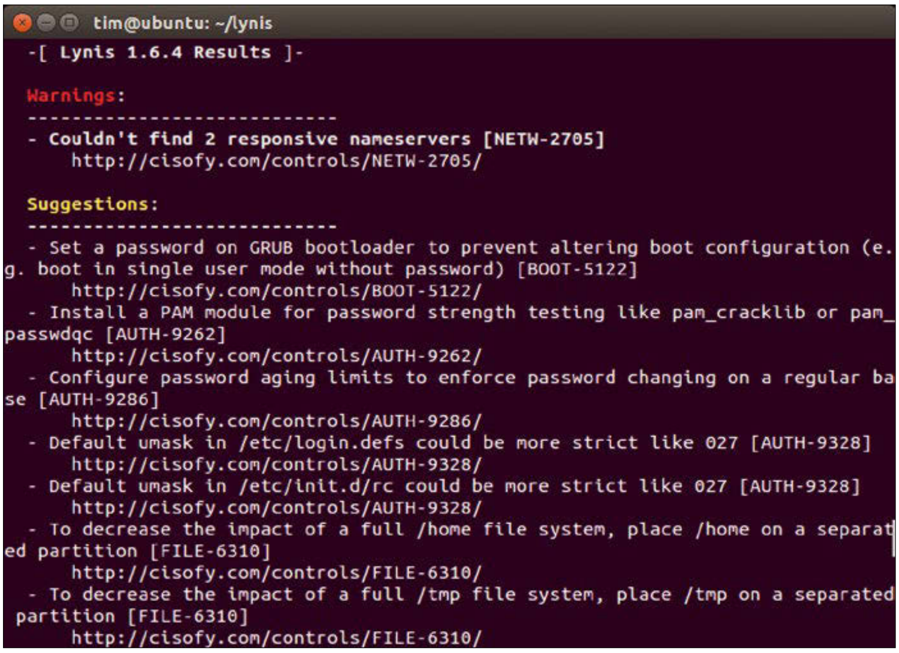
To identify problems that have crept in the back door, administrators will want to run Lynis regularly. The `--cronjob` option will help you execute the program on a regular schedule: If you run the tool with this parameter, it does not wait for user input and also removes all critical characters (special chars) from the input. In this way, you can run Lynis as a Cron job. Of course, the tool would overwrite the logfile and report file, `/var/log/lynis.log` and `/var/log/lynis-report.dat`, in each run. You can prevent Lynis from overwriting the logfiles by running Cron using a self-programmed wrapper script. Listing 1 contains a simple example. In the `extras/systemd` subdirectory of the Lynis archive, you will also find two files for Systemd: You can use the `service` file to enable Lynis automatically on system startup; The `timer` file launches the tool once a day. In both cases, it makes sense to copy Lynis to a directory such as `/usr/local/lynis`.

Using Individual Scans

If so desired, Lynis will also perform predefined tests. The approach is somewhat convoluted, however: Each test has a unique, cryptic ID. For example, Lynis refers to the test against the iptables kernel module as FIRE-4511. To find out which ID belongs to which test, let Lynis perform all the tests once, and then take a look at the `/var/log/lynis.log` file. It lists the IDs to match the tests.

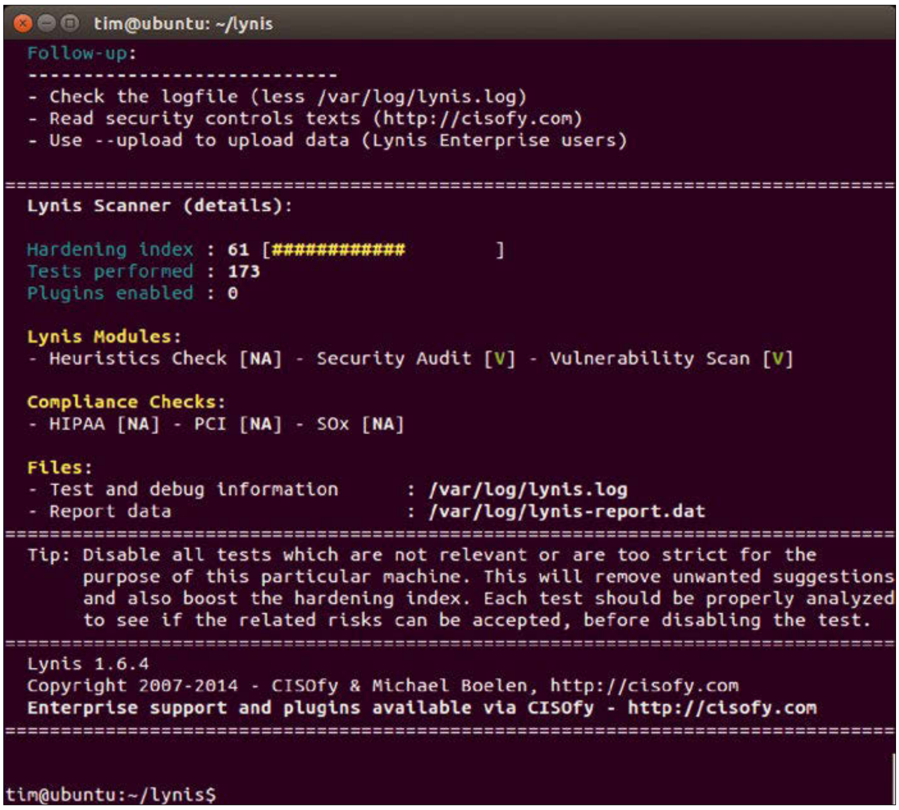
Another source of information are the shell scripts in the `include` subdirectory whose filenames start with `test_`. The scripts contain the tests themselves; for example the `tests_firewall` script groups all the tests relating to the firewall. The test IDs are included as comments in the script. Once you have determined the IDs for the desired tests, add them as a blank-separated list following the `--tests` parameter. As an example, the following command would execute the FIRE-4511 and AUTH-9226 tests:

```
./lynis -Q --tests "FIRE-4511 AUTH-9226"
```



```
tim@ubuntu: ~/lynis
  -[ Lynis 1.6.4 Results ]-

  Warnings:
  ----------------------------
  - Couldn't find 2 responsive nameservers [NETW-2705]
      http://cisofy.com/controls/NETW-2705/

  Suggestions:
  ----------------------------
  - Set a password on GRUB bootloader to prevent altering boot configuration (e.
g. boot in single user mode without password) [BOOT-5122]
      http://cisofy.com/controls/BOOT-5122/
  - Install a PAM module for password strength testing like pam_cracklib or pam_
passwdqc [AUTH-9262]
      http://cisofy.com/controls/AUTH-9262/
  - Configure password aging limits to enforce password changing on a regular ba
se [AUTH-9286]
      http://cisofy.com/controls/AUTH-9286/
  - Default umask in /etc/login.defs could be more strict like 027 [AUTH-9328]
      http://cisofy.com/controls/AUTH-9328/
  - Default umask in /etc/init.d/rc could be more strict like 027 [AUTH-9328]
      http://cisofy.com/controls/AUTH-9328/
  - To decrease the impact of a full /home file system, place /home on a separat
ed partition [FILE-6310]
      http://cisofy.com/controls/FILE-6310/
  - To decrease the impact of a full /tmp file system, place /tmp on a separated
 partition [FILE-6310]
      http://cisofy.com/controls/FILE-6310/
```

Figure 3: Lynis returns a list of proposed improvements that is fairly long on a freshly installed Ubuntu 14.10.



```
tim@ubuntu: ~/lynis
  Follow-up:
  ----------------------------
  - Check the logfile (less /var/log/lynis.log)
  - Read security controls texts (http://cisofy.com)
  - Use --upload to upload data (Lynis Enterprise users)

================================================================================
  Lynis Scanner (details):

  Hardening index : 61 [############         ]
  Tests performed : 173
  Plugins enabled : 0

  Lynis Modules:
  - Heuristics Check [NA] - Security Audit [V] - Vulnerability Scan [V]

  Compliance Checks:
  - HIPAA [NA] - PCI [NA] - SOx [NA]

  Files:
  - Test and debug information      : /var/log/lynis.log
  - Report data                     : /var/log/lynis-report.dat
================================================================================
  Tip: Disable all tests which are not relevant or are too strict for the
       purpose of this particular machine. This will remove unwanted suggestions
       and also boost the hardening index. Each test should be properly analyzed
       to see if the related risks can be accepted, before disabling the test.
================================================================================
  Lynis 1.6.4
  Copyright 2007-2014 - CISOfy & Michael Boelen, http://cisofy.com
  Enterprise support and plugins available via CISOfy - http://cisofy.com
================================================================================


tim@ubuntu:~/lynis$
```

Figure 4: Finally, Lynis outputs a "Hardening index." The higher the score, the better your system is protected against attacks – at least in the tool's opinion.

More Customization Options

In addition to the command-line parameters, a configuration (profile) file also controls the test process. By default, Lynis orients its actions on the specifications from the `default.prf` file supplied with the package. This file will probably be fine for most scenarios. You will typically only need to modify the file if you are using Lynis Enterprise. The structure and settings of the profile are explained in the comments it contains. You can tell Lynis to use your own profile from the `myprofile.prf` file by passing in the parameter `--profile myprofile.prf`.

You can add a number of tests to Lynis as plugins. A plugin is simply a normal shell script that begins with a couple of special comments containing details about its purpose. The script uses the functions provided by Lynis and resides in the plugins subdirectory. One example of a small plugin is the `plugins/custom_plugin.template` file. For Lynis to integrate and use the plugin, you need to register it in the profile. To do so, add a line stating `plugin=myplugin` for a plugin by the name of `myplugin`.

Conclusions

Lynis helps administrators identify vulnerabilities and problematic configurations. The tool is only capable of discovering problems that it is familiar with, of course. In particular, any homegrown scripts or other self-programmed software will be ignored. Additionally, Lynis only provides tips; it is up to the administrator to interpret the test results. Lynis is thus only one building block in your overall security system. ■■■

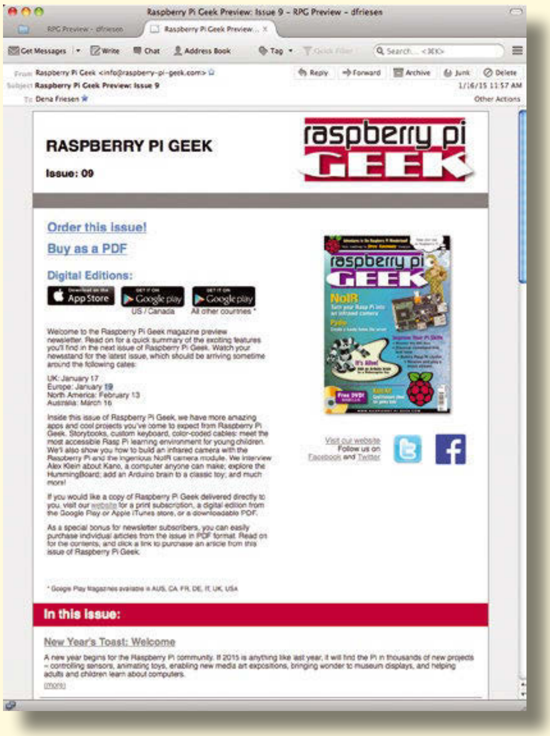
INFO

[1] Lynis *https://cisofy.com/lynis/*

[2] Lynis Enterprise: *https://cisofy.com/lynis-enterprise/*

[3] Lynis download page: *https://cisofy.com/download/lynis/*

[4] Lynis documentation: *https://cisofy.com/documentation/lynis/*



Raspberry Pi Geek Preview: Issue 9 – RPG Preview – dfriesen

From Raspberry Pi Geek <info@raspberry-pi-geek.com>
Subject Raspberry Pi Geek Preview: Issue 9
To Dena Friesen
1/16/15 11:57 AM

RASPBERRY PI GEEK

Issue: 09

Order this issue!

Buy as a PDF

Digital Editions:

App Store Google play US / Canada Google play All other countries *

Welcome to the Raspberry Pi Geek magazine preview newsletter. Read on for a quick summary of the exciting features you'll find in the next issue of Raspberry Pi Geek. Watch your newsstand for the latest issue, which should be arriving sometime around the following dates:

UK: January 17
Europe: January 19
North America: February 13
Australia: March 16

Inside this issue of Raspberry Pi Geek, we have more amazing apps and cool projects you've come to expect from Raspberry Pi Geek. Storybooks, custom keyboard, color-coded cables: meet the most accessible Rasp Pi learning environment for young children. We'll also show you how to build an infrared camera with the Raspberry Pi and the ingenious NoIR camera module. We interview Alex Klein about Kano, a computer anyone can make; explore the HummingBoard; add an Arduino brain to a classic toy; and much more!

If you would like a copy of Raspberry Pi Geek delivered directly to you, visit our website for a print subscription, a digital edition from the Google Play or Apple iTunes store, or a downloadable PDF.

As a special bonus for newsletter subscribers, you can easily purchase individual articles from the issue in PDF format. Read on for the contents, and click a link to purchase an article from this issue of Raspberry Pi Geek.

Visit our website
Follow us on
Facebook and Twitter

* Google Play Magazines available in AUS, CA, FR, DE, IT, UK, USA

In this issue:

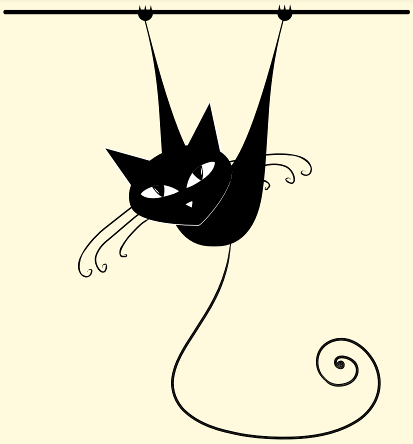
New Year's Toast: Welcome

A new year begins for the Raspberry Pi community. If 2015 is anything like last year, it will find the Pi in thousands of new projects – controlling sensors, animating toys, enabling new media art expositions, bringing wonder to museum displays, and helping adults and children learn about computers.

[more]

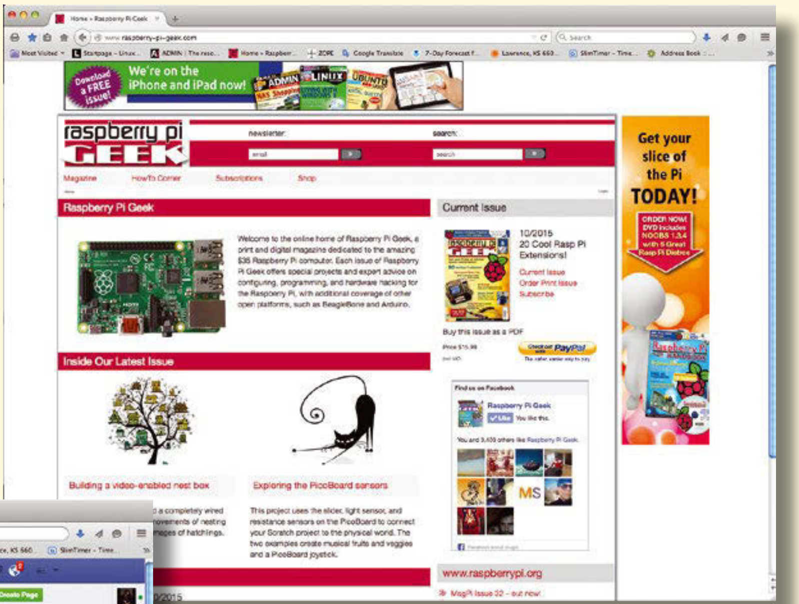
Sign up for our newsletter

www.raspberry-pi-geek.com/mc/subscribe



Visit us online

www.raspberry-pi-geek.com



Home » Raspberry Pi Geek
www.raspberry-pi-geek.com

Download a FREE issue! We're on the iPhone and iPad now!

newsletter: email
search: search

Magazine HowTo Corner Subscriptions Shop

Raspberry Pi Geek

Welcome to the online home of Raspberry Pi Geek, a print and digital magazine dedicated to the amazing $35 Raspberry Pi computer. Each issue of Raspberry Pi Geek offers special projects and expert advice on configuring, programming, and hardware hacking for the Raspberry Pi, with additional coverage of other open platforms, such as BeagleBone and Arduino.

Current Issue

10/2015
20 Cool Rasp Pi Extensions!
Current Issue
Order Print Issue
Subscribe

Buy this issue as a PDF
Price $15.99

Inside Our Latest Issue

Building a video-enabled nest box

...d a completely wired ...movements of nesting ...mages of hatchlings.

Exploring the PicoBoard sensors

This project uses the slider, light sensor, and resistance sensors on the PicoBoard to connect your Scratch project to the physical world. The two examples create musical fruits and veggies and a PicoBoard joystick.

Find us on Facebook
Raspberry Pi Geek
Like You like this.
You and 3,406 others like Raspberry Pi Geek.

www.raspberrypi.org
MagPi Issue 32 – out now!

Get your slice of the Pi TODAY!

Like us on Facebook

www.facebook.com/RasPiGeek

Raspberry Pi Geek
I won't stop until all the world knows about this fabulous creation!

Raspberry Pi Geek
Magazine

Liked Following Message

Timeline About Photos Likes Videos

PEOPLE
3,405 likes
Brian Osborn, Paul Brown and 5 others like this.
Invite your friends to like this Page

ABOUT
Raspberry Pi Geek is the first commercial print magazine dedicated to Raspberry Pi and the open hardware revolution. Each issue offers special projects and...
READ MORE
http://raspberry-pi-geek.com/
Suggest Edits

PHOTOS

Post Photo / Video
Write something on this Page...

Raspberry Pi Geek via Maker Faire
March 26 at 3:40am

Maker Faire Bay Area 2015: What Will You Find at the Faire?

Create Page
Recent
2014
2013
Sponsored
Trendy tops for spring fro...
www.zulily.com
Spring tops under $20!

Follow us on Twitter

@RasPi_Geek

Home Notifications Messages Discover Tweet

TWEETS 6 FOLLOWING 28 FOLLOWERS 85 Edit profile

Raspberry Pi Geek
@RasPi_Geek
Raspberry Pi Geek is the first commercial print magazine dedicated to Raspberry Pi and the open hardware revolution.
Lawrence, KS
raspberry-pi-geek.com

Tweets Tweets & replies

Kanawha Home Bay
@RasPi_Geek thanks for the new magazine get it while you can FREE DVD with every issue

Turn your Pi into an infrared remote control device!

Who to follow
Canon USA Corp. Follow
LinkedIn Follow
Atmel Corporation Follow

Trends

The latest ad tracking tricks and what to do about them

On the Canvas

We'll tell you about some powerful new ad tracking techniques and how you can stop them. *By Erik Bärwaldt*

Ad networks and companies are using increasingly sophisticated methods to track web surfers and spy on user behavior. However, the free web browser Firefox, in particular, makes it hard for these unabashed spies: various extensions block and remove standard cookies, web pixels, and well-hidden LSO cookies (also known as Flash cookies [1]).

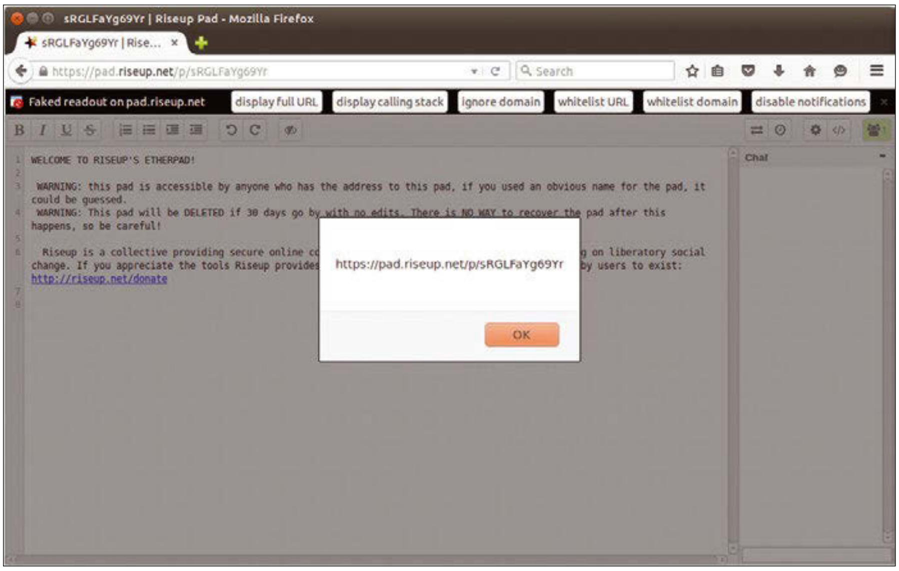
A young technology known as canvas fingerprinting does not require tricks like web pixels and LSO cookies and relies instead on standard HTML5 and JavaScript to help data grabbers track user behavior. In many cases, you can even accurately identify users. Because canvas fingerprints do not rely on additional data such as cookies on the system, conventional prevention methods fail.

Evercookies [2] are an older, but also increasingly popular, technique for spying on unsuspecting surfers. This article takes a close look at canvas fingerprinting and Evercookies and offers some options for how to stop these powerful tracking techniques.

Fingerprints

Almost all modern web browsers have supported the standardized HTML5 page description language since 2014. With its advanced commands and features, HTML5 gives programmers the ability to generate dynamic graphics. The canvas

Lead Image © Boyan Dimitrov, 123RF.com


sRGLFaYg69Yr | Riseup Pad - Mozilla Firefox

https://pad.riseup.net/p/sRGLFaYg69Yr

Faked readout on pad.riseup.net

```
WELCOME TO RISEUP'S ETHERPAD!

 WARNING: this pad is accessible by anyone who has the address to this pad, if you used an obvious name for the pad, it
could be guessed.
 WARNING: This pad will be DELETED if 30 days go by with no edits. There is NO WAY to recover the pad after this
happens, so be careful!

  Riseup is a collective providing secure online co                                       g on liberatory social
change. If you appreciate the tools Riseup provides                                       by users to exist:
http://riseup.net/donate
```

https://pad.riseup.net/p/sRGLFaYg69Yr

Figure 1: CanvasBlocker alerting on accessing a web page.

element of the command set identifies a region in which JavaScript can draw. You can also use the canvas element to call out, position, and scale text or graphics in the PNG, GIF, or JPEG format.

To create a clearly identifiable fingerprint of each surfer, canvas technology uses the fact that images and text in the canvas elements are displayed differently depending on the operating system, the web browser, the installed fonts, the graphics hardware, and the deployed drivers. Also, browser data such as the language, time zone, color depth, browser ID, and installed plugins vary from system to system.

Invisible graphics are output as a data URL, after injecting a hidden canvas element into a web page, and the script generates a hash value. When the surfer visits the same website with the same browser again, the tracker generates the same hash value again given an unchanged configuration.

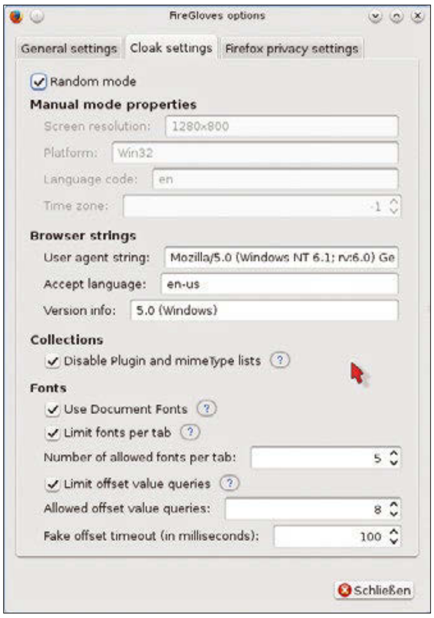
Thus, the script can very reliably identify the user. To track the user, ad networks place the same hidden canvas element on several websites and can then clearly identify users based on the same hash value.

The hit rate is particularly high for legacy desktop PCs with their extensive configuration options and variety of hardware components, operating systems, desktops, web browsers, and applications. The resulting large number of possible combinations translates to a similarly high rate of unique identification. Canvas fingerprinting works less successfully on mobile devices, such as smartphones or tablets, which are largely identical in terms of hardware and software, because the dynamically generated graphics only exhibit minor differences.

Redundant Cookies

Evercookies also use JavaScript to infest a computer system. In contrast to traditional cookies and Flash cookies, they use the web browser's individual storage technologies in a variety of combinations to nest multiple times in different locations. The history, browser cache, various HTML5 attributes – such as session, local, and global storage – as well as Silverlight Isolated Storage are all used to store Evercookies.

It is thus very hard to remove these pests completely from the system. If the user or a browser extension automatically deletes Evercookies in just some of these locations, they can be reconstructed from the remaining cookies.


FireGloves options

☑ Random mode

Manual mode properties

Screen resolution: 1280x800
Platform: Win32
Language code: en
Time zone: -1

Browser strings

User agent string: Mozilla/5.0 (Windows NT 6.1; rv:6.0) Ge
Accept language: en-us
Version info: 5.0 (Windows)

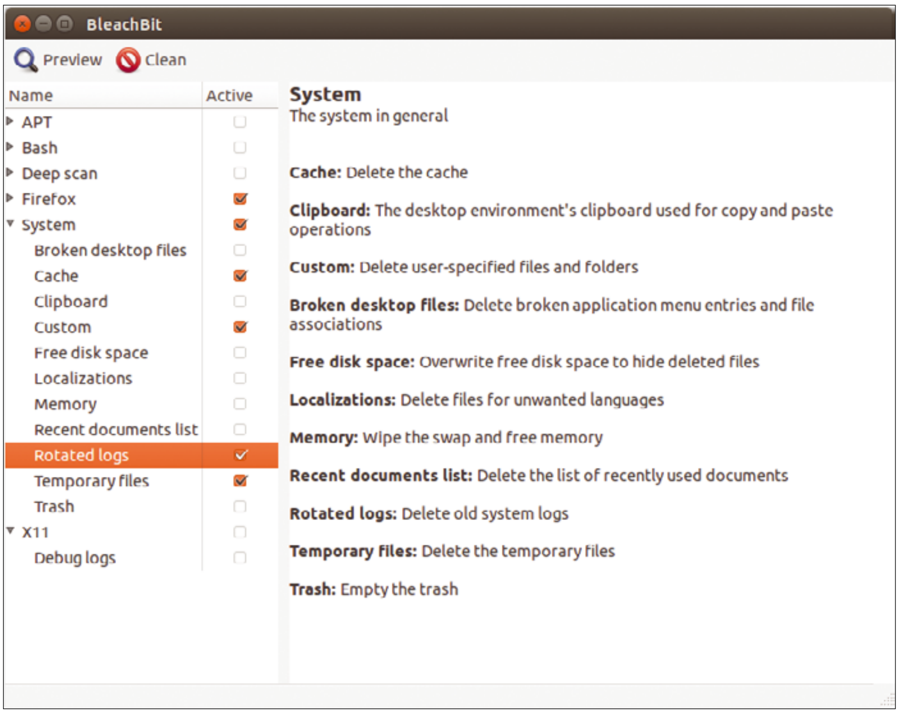
Collections

☑ Disable Plugin and mimeType lists

Fonts

☑ Use Document Fonts
☑ Limit fonts per tab
Number of allowed fonts per tab: 5
☑ Limit offset value queries
Allowed offset value queries: 8
Fake offset timeout (in milliseconds): 100

Figure 2: In this FireGloves dialog, you define how the plugin should spoof canvas elements.


BleachBit

Name	Active
▸ APT	□
▸ Bash	□
▸ Deep scan	□
▸ Firefox	☑
▾ System	☑
Broken desktop files	□
Cache	☑
Clipboard	□
Custom	☑
Free disk space	□
Localizations	□
Memory	□
Recent documents list	□
Rotated logs	☑
Temporary files	☑
Trash	□
▾ X11	□
Debug logs	□

System
The system in general

Cache: Delete the cache

Clipboard: The desktop environment's clipboard used for copy and paste operations

Custom: Delete user-specified files and folders

Broken desktop files: Delete broken application menu entries and file associations

Free disk space: Overwrite free disk space to hide deleted files

Localizations: Delete files for unwanted languages

Memory: Wipe the swap and free memory

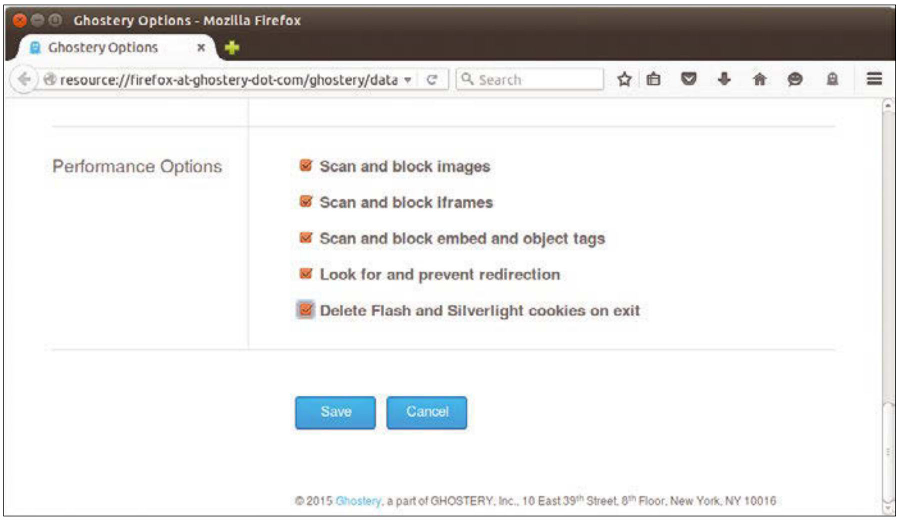
Recent documents list: Delete the list of recently used documents

Rotated logs: Delete old system logs

Temporary files: Delete the temporary files

Trash: Empty the trash

Figure 3: Bleachbit cleans the system, removing unnecessary ballast with just a few mouse clicks.


Ghostery Options - Mozilla Firefox
Ghostery Options
resource://firefox-at-ghostery-dot-com/ghostery/data
Search
Performance Options
Save
Cancel

☒ Scan and block images
☒ Scan and block iframes
☒ Scan and block embed and object tags
☒ Look for and prevent redirection
☒ Delete Flash and Silverlight cookies on exit

© 2015 Ghostery, a part of GHOSTERY, Inc., 10 East 39th Street, 8th Floor, New York, NY 10016

Figure 4: The Firefox Ghostery add-on deletes cookies from plugins such as Silverlight and Flash.

Thus, the usual browser extensions remain largely ineffective.

Detection Mechanisms

In as-delivered state, none of the popular web browsers can detect, remove, or block canvas fingerprints or Evercookies. Only the Tor Browser [3] emits a warning message if you call a web page that contains a canvas script, and it asks whether the browser should run or block the script. Additionally, canvas scripts presented to the Tor browser are not allowed to extract the implemented image data by default.

For Firefox, only the CanvasBlocker [4] add-on offers the ability to detect canvas call stacks (Figure 1). For Chrome/Chromium, there is CanvasFingerprintBlock [5], an add-on with a similar function. Like the Tor Browser, CanvasBlocker can block all or selected canvas elements in combination with Firefox. If the pop-up messages about discovered fingerprints at the top of the browser window disturb you, simply switch them off.

False Alarm

Strangely enough, the very popular free content management system WordPress introduced mechanisms in Version 4.2 that trigger false alarms in both the Tor browser and CanvasBlocker: The built-in emojis are to blame for this problem [6]. WordPress bundles the corresponding JavaScript code into a canvas element in the header of the web page without asking the user, so that the tracking tools trigger an alarm.

The WordPress CMS integrates this code into the header of the delivered pages, even if no emojis are used on the relevant website. To remove the typically unneeded canvas element from the header and maintain clean code, plugins such as Disable Emojis [7] are now available to WordPress developers.

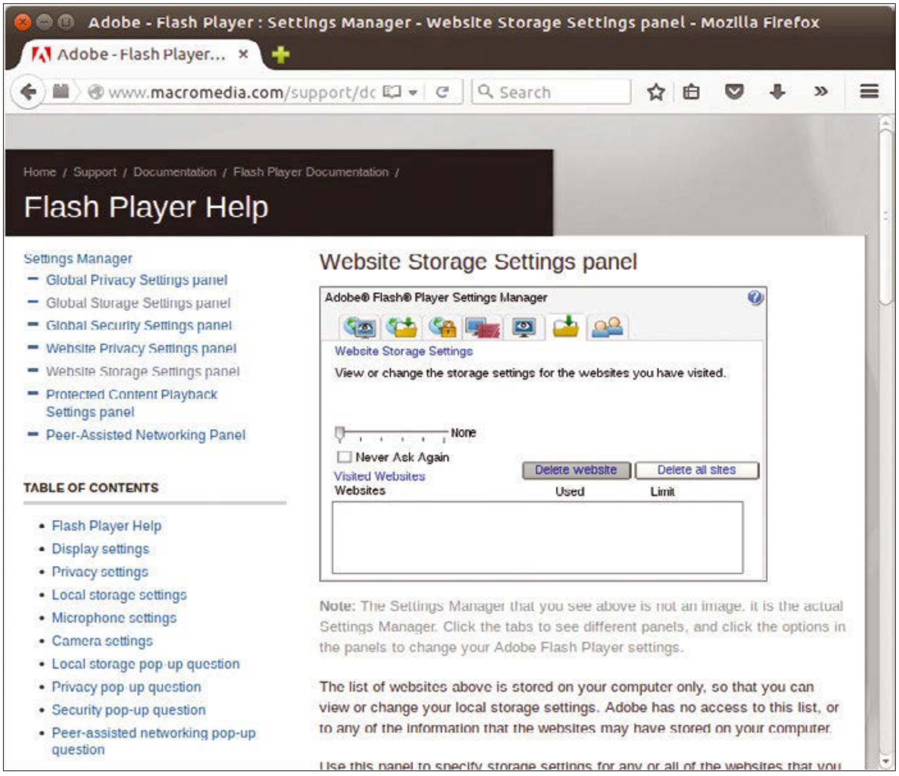
FireGloves

Because canvas elements and Evercookies are based on JavaScript, you can disable these pests by simply disabling JavaScript. Disabling JavaScript requires only one quick change in the browser's onboard toolkit, but removing JavaScript means many websites are no longer correctly rendered and functions and input are no longer possible. A better option is the Firefox FireGloves extension, which outmaneuvers any canvas fingerprinting detection mechanisms.

FireGloves is available in the official Mozilla add-on repository, but you can download it from an external website [8]. The XPI file of the extension is then installed via the *Install addon from file…* dialog in Firefox. Open the Add-on Manager and click the wrench icon at the top right of the window. After the installation, set up the add-on via the settings.

To make canvas fingerprints useless, the extension does not simply block all the canvas elements, but rather returns incorrect values for the browser and system parameters to the tracker. FireGloves randomly generates and updates incorrect parameters on a regular basis, and fingerprinting draws a blank.

However problems with displaying pages in the browser can occur when using FireGloves. For example, depending on which parameters the add-on uses, texts can appear in unusual fonts. In such cases, a simple mouse click on the FireGloves icon top right in the Firefox address bar usually does the


Adobe - Flash Player : Settings Manager - Website Storage Settings panel - Mozilla Firefox
Adobe - Flash Player...
www.macromedia.com/support/dc
Search
Home / Support / Documentation / Flash Player Documentation /
Flash Player Help
Settings Manager
Global Privacy Settings panel
Global Storage Settings panel
Global Security Settings panel
Website Privacy Settings panel
Website Storage Settings panel
Protected Content Playback Settings panel
Peer-Assisted Networking Panel
TABLE OF CONTENTS
Flash Player Help
Display settings
Privacy settings
Local storage settings
Microphone settings
Camera settings
Local storage pop-up question
Privacy pop up question
Security pop-up question
Peer-assisted networking pop-up question
Website Storage Settings panel
Adobe® Flash® Player Settings Manager
Website Storage Settings
None
Delete website
Delete all sites
Visited Websites
Websites
Used
Limit

View or change the storage settings for the websites you have visited.

□ Never Ask Again

Note: The Settings Manager that you see above is not an image. it is the actual Settings Manager. Click the tabs to see different panels, and click the options in the panels to change your Adobe Flash Player settings.

The list of websites above is stored on your computer only, so that you can view or change your local storage settings. Adobe has no access to this list, or to any of the information that the websites may have stored on your computer.

Use this panel to specify storage settings for any or all of the websites that you

Figure 5: Adobe offers an online tool for deleting content.

trick. The extension creates new parameters and thus modifies the appearance of the web page.

FireGloves summarizes some important web browser security settings in a simple options menu. You can access the dialog by right clicking on the blue *Info* icon next to the FireGloves glove in the Firefox address bar and selecting *Open preferences*. Alternatively, go to the browser's add-on list via the *Settings* button and look for the *FireGloves* line.

In the *Options* window, you will find three tabs where you can configure the desired settings. The first tab only enables the add-on on launching Firefox. In the second tab, *Cloak settings*, you can determine in detail what data you want FireGloves to deliver about your browser, your system, and ultimately, about you. This includes the option of enabling random mode, which randomly selects the parameters and thus optimally camouflages the system.

The third candidate, *Firefox privacy settings*, groups a number settings otherwise found in different dialogs in Firefox. In this way, you can easily harden your system, and without complicated searching in various option dialogs (Figure 2).

Second Line of Defense

Simple add-ons are not enough to render Evercookies completely harmless. The problem is that Evercookies sometimes use technologies such as Flash and Silverlight that work regardless of Firefox.

On Linux, however, you can rely on a tool like Bleachbit [9] to clean up the individual Evercookies locations after every session, without having to painstakingly configure a variety of add-ons. Bleachbit is available from the repositories of all popular distributions and is easily installed using your distro's package manager.

Bleachbit automatically determines at startup, which applications are available on your system and displays a list of contents to be deleted in the left pane of its program window. Check the buttons to the left of each entry in order to remove an item. Help for the individual deleted options is shown on the right side of the program window; you thus know at a glance whether or not it is a good idea to switch certain options (Figure 3).

To delete the Evercookies storage locations, it is a good idea to use Bleachbit's Firefox-specific options to clean up the *Profile data (Cookies)*, the *DOM memory*, the *Address history*, and the *Cache*. Additionally, you will also want to delete all Flash content to eradicate the infamous LSO cookies.

You can also use the Ghostery [10] Firefox add-on, which primarily blocks web pixels that also spy on a user's surfing behavior, to delete Flash and Silverlight cookies. See the configuration options in the *Advanced* tab (Figure 4).

Ideally, you would then do without the critical Flash and Silverlight plugins in the future and surf the web with a "clean" browser.

The Adobe Variant

An Adobe online service offers a different method for removing Flash cookies

from your system. Go to the Setting Manager on the Macromedia website [11], click the *Global Storage Settings* tab, and disable the options *Allow third-party Flash content to store data on your computer* and *Store common Flash components to reduce download times*. Also, delete any existing LSO cookies below *Website Storage Settings* (Figure 5).

Conclusions

Although the advertising industry is doing somersaults to spy on unsuspecting web surfers, free developers are investing at least as much time and energy to guarantee data protection even against highly complex spyware. You do not need to rely on multiple browser add-ons to remove annoying pests from your system, but you can redirect these intrusion attempts to a black hole with just a few clicks.

Whatever the circumstances, it is always advisable to keep the system clean with a combination of add-ons and the Bleachbit tool, because a cleaner system means fewer loopholes for Evercookies.

Canvas fingerprints can be effectively and easily misled using the Firefox FireGloves add-on. Therefore, the advertising industry will need to come up with somewhat more sophisticated mechanisms in the future to spy on users of free software. ■■■

INFO

[1] Flash cookie: *https://en.wikipedia.org/wiki/Local_shared_object*

[2] Evercookies: *https://en.wikipedia.org/wiki/Evercookie*

[3] Tor Browser: *https://www.torproject.org/projects/torbrowser.html.en*

[4] CanvasBlocker: *https://addons.mozilla.org/en-us/firefox/addon/canvasblocker/*

[5] CanvasFingerprintBlock: *https://chrome.google.com/webstore/detail/canvasfingerprintblock/ipmjngkmngdcdpmgmiebdmfbkcecdndc*

[6] Wordpress 4.2 “Powell”: *https://wordpress.org/news/2015/04/powell*

[7] Disable Emojis: *https://wordpress.org/plugins/disable-emojis*

[8] FireGloves: *http://fingerprint.pet-portal.eu/?menu=6*

[9] Bleachbit: *http://bleachbit.sourceforge.net/*

[10] Ghostery: *https://www.ghostery.com/our-solutions/ghostery-add-on*

[11] Deleting Flash cookies: *http://www.macromedia.com/support/documentation/en/flashplayer/help/settings_manager.html*

[12] headtrip.io GbR: *http://headtrip.eu* (in German)

[13] BetterPrivacy: *https://addons.mozilla.org/en-US/firefox/addon/betterprivacy/*

[14] Tails: *https://tails.boum.org*

INTERVIEW: Canvas Fingerprinting and Evercookies

Canvas fingerprinting and Evercookies are two relatively unknown methods for spying on the surfing habits of Internet users. We asked Florian Drechsler, eCommerce expert, web designer, and co-owner of headtrip.io GbR from Nuremberg, Germany, [12], for his assessment of future developments and how to best protect yourself as an Internet surfer.

Linux Magazine: *Canvas fingerprinting on web pages first attracted greater attention last summer, when researchers at the universities of Leuven and Princeton provided evidence of this tracking method on almost six percent of all surveyed web sites. Since then, public interest in this technique has again waned somewhat. Based on your experience, are there signs that canvas fingerprints are increasingly being used to identify surfers and their surfing habits on the Internet?*

Florian Drechsler: Definitely, yes. The registered percentage of affected websites at that time was attributed to a large extent to advertising service provider AddThis, who apparently used canvas fingerprints to deliver personalized ads. But, AddThis quickly responded to the criticism and removed the canvasing code. In my experience, canvas fingerprinting has definitely spread – simply because many eCommerce companies and advertising platforms see it as a possibility to boost conversion rates through personalized content.

LM: *The Tor Browser warns users about canvas fingerprints on many web pages. Often, also the Firefox CanvasBlocker extension indicates that canvas elements are trying to extract image files that could be used for spying on surfers. Analysis of the source code on most affected web pages show that the canvas code causing the alert was attributable to a small script introduced in WordPress 4.2 that checks to see whether emojis are available. Do such extensions that allow visitors to websites to be spied on cause any real danger?*

FD: The Emoji script itself is harmless. Instead, the danger lies in the fact that the user approves this innocuous usage of the canvas element, and thus allows other potentially malicious elements.

LM: *How can surfers tell, when they are notified of canvas fingerprints, whether those elements are used for tracking?*

FD: If you cannot analyze the code yourself, your only option – as is so often the case in Internet security – is to rely on common sense. To do this, however, you need to know how a canvas element works. Canvas elements are used by websites for drawing, say, 3D animations or for browser games. In case of doubt, you should block the canvas element and then try to use the site: Are you missing some elaborate graphics? If so, switch the canvas back on. But if the site works without a canvas element, then it was at least superfluous, or it was actually used to track users.

LM: *Evercookies are a tracking method that is as difficult to control as canvas fingerprints. How can I protect myself against Evercookies?*

FD: By installing the Firefox BetterPrivacy [13] extension, which deletes Flash cookies and runs the browser in private browsing mode. If you do not need plugins like Silverlight and Flash, you should turn them off – and not only because of the Evercookies. The safest method, however, is the use of a specially hardened Linux distribution such as Tails [14].

LM: *How do you see future developments: Are Evercookies and canvas fingerprints likely to spread?*

FD: The final version of HTML5 is now only a few months old, and it might take some time until all clients can use canvas elements at all. The more frequently canvas elements are used, the more attractive options for using canvas fingerprinting will become. Evercookies have been around for over five years and are still in active development. Other methods that allow storage of user data might also arise through exploiting new browser technologies. Online traders, in particular, benefit from Evercookies and canvas fingerprinting, which let them trace the surfing behavior of potential customers. I would assume this option is used by increasing numbers of eCommerce companies.

usenix
THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

UPCOMING EVENTS

Enigma
January 25–27, 2016, San Francisco, CA, USA
enigma.usenix.org

FAST '16: 14th USENIX Conference on File and Storage Technologies
February 22–25, 2016, Santa Clara, CA, USA
www.usenix.org/fast16

NSDI '16: 13th USENIX Symposium on Networked Systems Design and Implementation
March 16–18, 2016, Santa Clara, CA, USA
www.usenix.org/nsdi16

Co-located with NSDI '16:

CoolDC '16: USENIX Workshop on Cool Topics on Sustainable Data Centers
March 19, 2016, Santa Clara, CA, USA
Submissions due December 15, 2015
www.usenix.org/cooldc16

SREcon16
April 7–8, 2015, Santa Clara, CA, USA

USENIX ATC '16: 2016 USENIX Annual Technical Conference
June 22–24, 2016, Denver, CO, USA
Submissions due February 1, 2016
www.usenix.org/atc16

Co-located with USENIX ATC '16:

HotCloud '16: 8th USENIX Workshop on Hot Topics in Cloud Computing
June 20–21, 2016

HotStorage '16: 8th USENIX Workshop on Hot Topics in Storage and File Systems
June 20–21, 2016

SOUPS 2016: Twelfth Symposium on Usable Privacy and Security
June 22–24, 2016
www.usenix.org/soups2016

SREcon16 Europe
July 11–13, 2016, Dublin, Ireland

USENIX Security '16: 25th USENIX Security Symposium
August 10–12, 2016, Austin, TX, USA

Co-located with USENIX Security '16

WOOT '16: 10th USENIX Workshop on Offensive Technologies
August 8–9, 2016

CSET '16: 9th Workshop on Cyber Security Experimentation and Test
August 8, 2016

ASE '16: 2016 USENIX Workshop on Advances in Security Education
August 9, 2016

HotSec '16: 2016 USENIX Summit on Hot Topics in Security
August 9, 2016

OSDI '16: 12th USENIX Symposium on Operating Systems Design and Implementation
November 2–4, 2016, Savannah, GA, USA
Abstract registration due May 3, 2016
www.usenix.org/osdi16

LISA16
December 4–9, 2016, Boston, MA, USA

Do you know about the USENIX Open Access Policy?

USENIX is the first computing association to offer free and open access to all of our conferences proceedings and videos. We stand by our mission to foster excellence and innovation while supporting research with a practical bias. Your membership fees play a major role in making this endeavor successful.

Please help us support open access. Renew your USENIX membership and ask your colleagues to join or renew today!

www.usenix.org/membership

Stay Connected...

twitter.com/usenix
www.usenix.org/youtube
www.usenix.org/gplus
www.usenix.org/facebook
www.usenix.org/linkedin
www.usenix.org/blog

The sys admin's daily grind: ddrescue and DDRescue-GUI

Recovery Needed

Sometimes even sys admin Charly doesn't have a backup at hand; or, maybe it's ruined because the removed disk had corrupt data. Here, he offers some advice on how to handle the situation. *By Charly Kühnast*

Krrr, krrr ...! At least things are clear-cut when a hard disk gives up the ghost: You toss the offending disk, get a new one, and put the backup on it. However, those undead data media – that trick people into continuing working on them with no idea of the potential impact – are a real pain.

I recently determined that an SDHC card in my camera saves one out of 20 images (on average) as a colorful mess of pixels. I do know that memory cards give up the ghost sooner or later. However, I didn't realize that my camera could save to two cards simultaneously – a feature I stupidly didn't use. But, I'm all the wiser now.

What if really important data is stored on a haywire device that you just can't get rid of? This is where `ddrescue` [1] comes in. The tool is already quite ancient, but its developers look after it untiringly and adapt it to new types of data media. (It should not be confused with the even older `dd_rescue`.) Ddrescue is officially named GNU ddrescue; the packages on Debian and derivatives are therefore dubbed `gddrescue`.

The tool is included with many popular distributions. The first two letters subtly indicate a relationship with `dd`, and `ddrescue` actually generates a data medium's or partition's image. Unlike `dd`, however, it can't be stopped by read errors; instead, it stubbornly saves everything that it can get its teeth into.

Two-Speed Transmission

Administrators usually use ddrescue in two phases. The first phase involves creating an image with all the data that can be accurately read. In the following example, `/dev/sdd1` is a partition with read errors on a USB flash drive:

```
sudo ddrescue -n /dev/sdd1 ⮯
  /home/charly/stick.img logfile.log
```

The second, more time-consuming phase involves using the tool to sort through the faulty blocks and save as much data from them as possible. The command is just the same as before, except you leave out the `-n` parameter. In the wake of ddrescue, there is still an armada of other parameters that control the tool's behavior.

There is also a GUI [2] that you can use to make some quick, useful default settings. I installed it quickly on my test Ubuntu using these three steps:

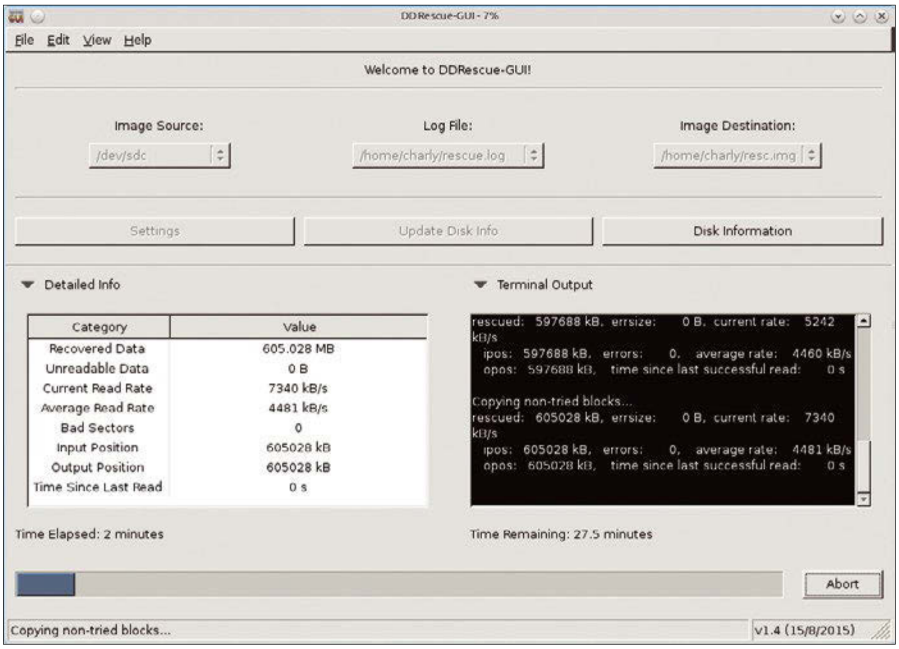
```
sudo add-apt-repository ppa:hamishmb/myppa
sudo apt-get update
sudo apt-get install -fym ddrescue-gui
```

As Figure 1 shows, the interface is businesslike and functional. The GUI sets the important parameters, but not all of them by far. Although I hope no one will need to use ddrescue permanently, the GUI is nevertheless a real help. ■■■

INFO

[1] ddrescue: *http://www.gnu.org/software/ddrescue/*

[2] DDRescue-GUI: *https://launchpad.net/ddrescue-gui*



Welcome to DDRescue-GUI!

Image Source: /dev/sdc
Log File: /home/charly/rescue.log
Image Destination: /home/charly/resc.img

Category	Value
Recovered Data	605.028 MB
Unreadable Data	0 B
Current Read Rate	7340 kB/s
Average Read Rate	4481 kB/s
Bad Sectors	0
Input Position	605028 kB
Output Position	605028 kB
Time Since Last Read	0 s

```
rescued:  597688 kB,  errsize:     0 B,  current rate:   5242
kB/s
   ipos:  597688 kB,   errors:      0,   average rate:   4460 kB/s
   opos:  597688 kB,   time since last successful read:     0 s

Copying non-tried blocks...
rescued:  605028 kB,  errsize:     0 B,  current rate:   7340
kB/s
   ipos:  605028 kB,   errors:      0,   average rate:   4481 kB/s
   opos:  605028 kB,   time since last successful read:     0 s
```

Time Elapsed: 2 minutes

Time Remaining: 27.5 minutes

Copying non-tried blocks...

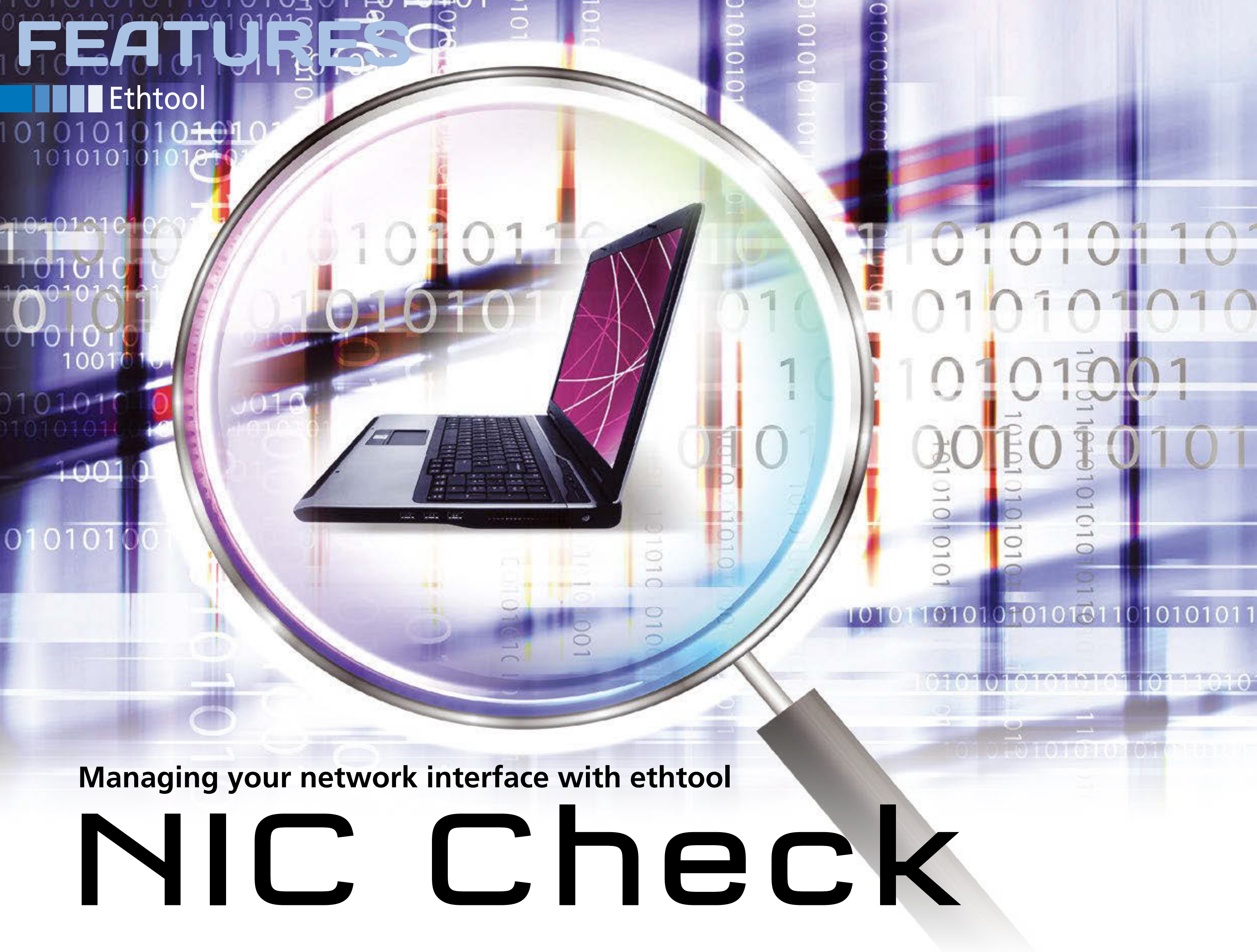
v1.4 (15/8/2015)

Figure 1: The genuinely helpful front end for ddrescue is DDRescue-GUI, which graphically implements the important parameters.

CHARLY KÜHNAST

Charly Kühnast is a Unix operating system administrator at the Data Center in Moers, Germany. His tasks include firewall and DMZ security and availability. He divides his leisure time into hot, wet, and eastern sectors, where he enjoys cooking, freshwater aquariums, and learning Japanese, respectively.

FREE DVD
sabayon 64-bit
mageia 32-bit
YOU AND HTTP/2
Exploring the new web standard
LINUX MAGAZINE
JUST RELEASED!
DECEMBER 2015
HTTP/2
Why the new web is different and why it matters
JMeter
Cool tool for testing your web server
ARP Protocol Attacks
Build Your Own PPA
Package your apps and share them with other users
LibreOffice 5
What's new and better in the latest release
FlexGet
Download Internet content automatically
ss
Manage sockets and connections
Cryptonify
Keep track of online passwords and login info
WWW.LINUX-MAGAZINE.COM
GET IT NOW!
SAVE TIME ON DELIVERY WITH OUR PDF EDITION
ISSUE 181
DEC 2015
DVD ROM



Managing your network interface with ethtool

NIC Check

If ping won't solve your network configuration issues, try ethtool, a powerful utility that lets you manage configuration settings for your network interface card. *By Chris Binnie*

The sheer volume of traffic on a super-busy and ever-changing production network makes any oddities very tricky to diagnose. Many of the more serious networking headaches are devastatingly disruptive to business operations. Additionally, the constantly changing nature of busy networks, coupled with the 24/7 attacks of varying competency, mean you need to know your stuff and maintain eagle-eyed diligence in order to catch issues.

If you're like me, though, you probably try to solve these kinds of issues in a simple and elegant way. To alter the network settings on a NIC (Network Interface Card), you have almost certainly used the now deprecated `ifconfig` command and its successor the `ip` command.

Another useful utility you can use to troubleshoot network problems is `ethtool`. The malleable ethtool isn't concerned with IP addresses, VLANs, and subnets. Instead, ethtool lets you manage and configure software drivers and hardware settings that control network interfaces.

If you want to know what the main network interface, `eth0`, is up to at this moment, you can enter the following command as root:

```
ethtool eth0
```

The output is shown in Listing 1.

As you can see, the output in Listing 1 provides information on the network configuration, including the speed of the network card, as well as details such as the `MDI-X` setting, which allows a network interface to figure out if a "straight-through" or "crossover" cable might be necessary to complete the connection.

Probably of greatest benefit of the information presented in Listing 1 is that you can determine the network card settings without breaking a sweat. The following facts are quite useful for understanding the network connection:

```
Speed: 10000Mb/s
Duplex: Full
Auto-negotiation: off
Link detected: yes
```

Note that this system is managing to speak to the next device in the chain at a massive 10Gb/s. That device might be some kind of networking kit, such as a switch.

This super-fast speed occurs simultaneously in both directions because it's a full-duplex link, which is just what I'm looking for to achieve the highest data transfers speed possible on the NIC. Finally, the link is detected as being `up`, and the settings are forced onto the card, rather than acquired through the occasionally-problematic auto-negotiation.

Changes

In addition to telling you what the NIC is doing, ethtool also lets you change the configuration settings. If you want to

Lead Image © Sean Gladwell, Fotolia.com

make sure your precious primary interface doesn't try to connect to the upstream switch using auto-negotiation, enter the following:

```
# ethtool -s eth0 autoneg off
```

The `-s` option specifies that you're making a change. In case it's a stumbling block in the future, the `-s` switch is needed to apply any new settings that you make, so feel free to use it readily (but with care). The other parts of the command should be self-explanatory.

The capitalized version of the `-s` option, the uppercase `-S`, is nothing like its baby brother and is used to request some exceptionally useful statistics.

Listing 2 shows the abbreviated output of the `ethtool -S` command (the full output is actually about two thirds longer). The statistics shown in Listing 2 are invaluable when troubleshooting a malfunctioning NIC. I've even created a `cron` job just to email these verbose statistics to me on a daily basis. That way, when I'm doing my preflight checks in the morning, I can scan the error sections for peace of mind that a change I made previously is working as expected.

The most common use of ethtool is to alter the speed and duplex of a network interface. Imagine the scenario where your 100Mbps NIC is causing all sorts of reliability issues and you desperately need it to continue working, even if that means working more slowly. One thing to try is to switch off auto-negotiation, but you can also consider forcing the link speed to much slower by dropping it all the way down to 10Mbps:

```
# ethtool -s eth0 speed 10 autoneg off
```

Or, throw duplex preference into the mix:

```
# ethtool -s eth0 speed 10 ↩
  duplex half autoneg off
```

This command slows the link down to its minimum and also disables auto-negotiation, giving a better chance for some form of connectivity, even if it is much slower.

If you can access the machine out-of-band (dial-up through "mgetty," for instance, is perfectly suitable for this kind of test), try a few settings and revert them quickly if they prove to be unsuitable. This type of troubleshooting is often the fastest approach if you are debugging the NIC for a server located many miles away in a data center.

Persistence

The examples I've described so far are not persistent changes and will disappear following a reboot.

On RHEL and its associative flavors, you can add a line to the end of your networking config file (which is `/etc/sysconfig/network-scripts/ifcfg-eth0` for the `eth0` interface):

```
ETHTOOL_OPTS="speed 10 ↩
  duplex half autoneg off"
```

You could then (carefully, to avoid being locked out) stop and start your NIC (preferably by using `/etc/init.d/network restart` or `service network restart` in the first instance) with a natty little command such as:

```
# ifdown eth0 && ifup eth0
```

You can achieve a similar result on Debian-based systems, although it's a pretty nasty hack that's about as graceful as an elephant high-diving into a swimming pool full of custard.

Append the following lines into the long-serving Unix-like hidden file `/etc/rc.local` , and your NIC settings should survive a reboot. Obviously, you can adjust them to your needs.

```
ethtool -s eth0 autoneg off
ethtool -s eth0 speed 10
ethtool -s eth0 duplex half
```

With different versions, your mileage

LISTING 1: Standard Info for etho

```
# ethtool eth0

Settings for eth0:
        Supported ports: [ TP ]
        Supported link modes:   1000baseT/Full
                                10000baseT/Full
        Supported pause frame use: No
        Supports auto-negotiation: No
        Advertised link modes:  Not reported
        Advertised pause frame use: No
        Advertised auto-negotiation: No
        Speed: 10000Mb/s
        Duplex: Full
        Port: Twisted Pair
        PHYAD: 0
        Transceiver: internal
        Auto-negotiation: off
        MDI-X: Unknown
        Supports Wake-on: uag
        Wake-on: d
        Link detected: yes
```

LISTING 2: Abbreviated -S Statistics Request

```
NIC statistics:
     Tx Queue#: 0
       TSO pkts tx: 727069
       TSO bytes tx: 2238017010
       ucast pkts tx: 3184293
       ucast bytes tx: 3446010007
       mcast pkts tx: 0
       mcast bytes tx: 0
       bcast pkts tx: 121
       bcast bytes tx: 5082
       pkts tx err: 0
       pkts tx discard: 0
       drv dropped tx total: 0
          too many frags: 0
          giant hdr: 0
          hdr err: 0
          tso: 0
       ring full: 0
       pkts linearized: 0
       hdr cloned: 0
       giant hdr: 0
```

may well vary, so be prepared for some potential head-scratching.

Another approach is to write a custom `init` script that will run at boot time (and then adjust it as needed when you upgrade to a `systemd`-based version). Also, a more suitable way might be to insert the config into the file `/etc/network/interfaces`.

Once you've found the section where the pertinent NIC is mentioned (I'll stick with the trusty `eth0`), you can append the `pre-up` line, as shown below, adjusting it as needed:

```
auto eth0
iface eth0 inet static
pre-up /sbin/ethtool -s eth0 ↩
  speed 10 duplex half autoneg off
```

In the worst case, you could always use the nasty `/etc/rc.local` fix, which has been a bone of contention since time immemorial. If there were version headaches, I would cobble together a simple `init` script instead, which is only one rung higher up on the laziness ladder.

More Options

I've barely scratched the surface of the useful reporting that the excellent ethtool can offer. My favorite part about ethtool is you can easily combine the output into a script in order to display the current status of a network card. I can imagine some of these configurable settings also being easily changeable within a web interface, along the lines of GUIs, such as Webmin, for example. Listing 3 shows ethtool with the `-i` switch, which reveals driver details for the NIC.

You can use ethtool for lots of other configuration changes. The Fedora documentation provides a useful summary of some of the most important options [2].

LISTING 3: Viewing NIC Driver Details

```
# ethtool -i eth0

driver: intel-fyvm
version: 1.4.12.0-k-NAPI
firmware-version:
bus-info: 0000:0b:00.0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: yes
supports-priv-flags: no
```

MII-TOOL

An older tool called mii-tool also allows the user to query NIC settings. Mii-tool is a precursor to ethtool, and many consider it obsolete, but it is still effective on older systems where ethtool isn't available.

If you are going to try mii-tool, be aware of the caveat in the "NOTE" section of the mii-tool man page:

"This program is obsolete. Valid media are only 100baseT4, 100baseTx-FD,100baseTx-HD, 10baseT-FD and 10baseT-HD ethernet cards. For replacement, check ethtool."

In layman's terms, the note means you can't count on mii-tool to help you if the network speed is higher than 100Mbps.

To drop link up/down messages to the Syslog daemon:

```
# mii-tool -lw
```

To force a speed setting:

```
# mii-tool --force 10baseT-HD
```

`HD` stands for "half duplex." `FD` cranks up the setting to full blast.

You might get lost in the configuration parameters available from mii-tool and need to reset the config back to defaults (hardware settings only):

```
# mii-tool --reset
```

The Fedora page offers a useful setting for how to quickly change your MAC address; ethtool refers to this type of address as a *physical address* and uses the following syntax:

```
--phyad HEX-VALUE devname
```

Simply replace `HEX-VALUE` with a hexadecimal MAC address like `00:20:12:1a:9f:de` and replace `devname` with `eth0`.

TSO

Another gotcha I've come across in the past is called TCP Segmentation Offload (known as TSO), which is used for outbound, egress traffic. TSO is well-intentioned and is designed to improve the throughput performance of a NIC while helping to cut down on the CPU hit for processing high levels of traffic.

The hardware helps to introduce a buffer that can then have its contents split up into segments. In short, TSO has control and may change how your TCP traffic flows. This feature works well until something breaks its functionality and you need to disable it quick smart. You can disable TSO with:

```
# ethtool -K eth0 tso off
```

Conversely, if you want to re-enable it, you can use the command:

```
# ethtool -K eth0 tso on
```

But, there's more. In the following command:

```
# ethtool -K eth0 tx off ↩
  sg off tso off
```

the `tx` relates to pause functionality for transmitted, egress data. The `sg` stands for *scatter-gather*. From what I can tell, *scatter-gather* is the act of pulling together data from different places and reformatting it so that you can output it as one data stream.

For a network card to support TSO, both checksumming and scatter-gather functionality must be present. Switching off the ability to *pause* traffic, as well as `sg` and TSO, is a good starting point for debugging. You have `ufo` (UDP Fragmentation) and `gso` (Generic Segmentation) options as well.

Incidentally, TSO's counterpart for ingress traffic is something called LRO (Large Receive Offload), which uses a very similar premise and the `lro` moniker as its syntax. Some reports say using LRO on Linux routers is a very bad idea that will cause significant performance headaches, so watch out with LRO if you're forwarding traffic.

Flow Control

Another common networking issue is *flow control*. In its first incarnation, flow control was defined in IEEE 802.3x, and later in IEEE 802.1Qbb, which describes "priority-based flow control."

Flow control works clevely by offering a helping hand to the part of the network link that has too much work to do. When the network is struggling, flow control allows a brief pause so that traffic can catch up. Unfortunately, in some situations, overall network speed can be severely affected if the clever flow control misbehaves.

One such nightmare scenario might be using Gigabit switches, Gigabit servers, and 100Mbps client machines. Badly behaving flow control might force all devices, Gigabit or otherwise, all the way down to 100Mbps.

You'll need to have a network card that supports pause functionality to adjust these settings. The most basic type of pause functionality lets you pause all traffic on a link; a newer version also lets you prioritize per traffic class. You can query your card with the `-a` option, which is the same as `--show-pause`:

```
# ethtool -a eth0
```

```
Pause parameters for eth0:
Autonegotiate:  on
RX:             on
TX:             on
```

For this NIC, all is well with hardware support.

You can disable this sometimes-pesky pause functionality with the following syntax:

```
# ethtool -a|--pause devname ↲
  [rx on|off] [tx on|off] ↲
  [autoneg on|off]
```

In other words, an example to switch off both ingress and egress "pause" functionality might look like the following:

```
# ethtool --pause eth0 ↲
  rx off tx off autoneg off
```

Some users contend that making sure the auto-negotiation is switched off will ensure your settings are applied correctly. You can simply run:

```
# ethtool -a eth0
```

to check whether your new settings have taken effect.

Unconscious

At some point in the future, you'll be sitting impatiently, blaming a piece of software for your woes and the resulting bad mood your boss is in. Lo and behold the issue might be a network problem you can diagnose with ethtool.

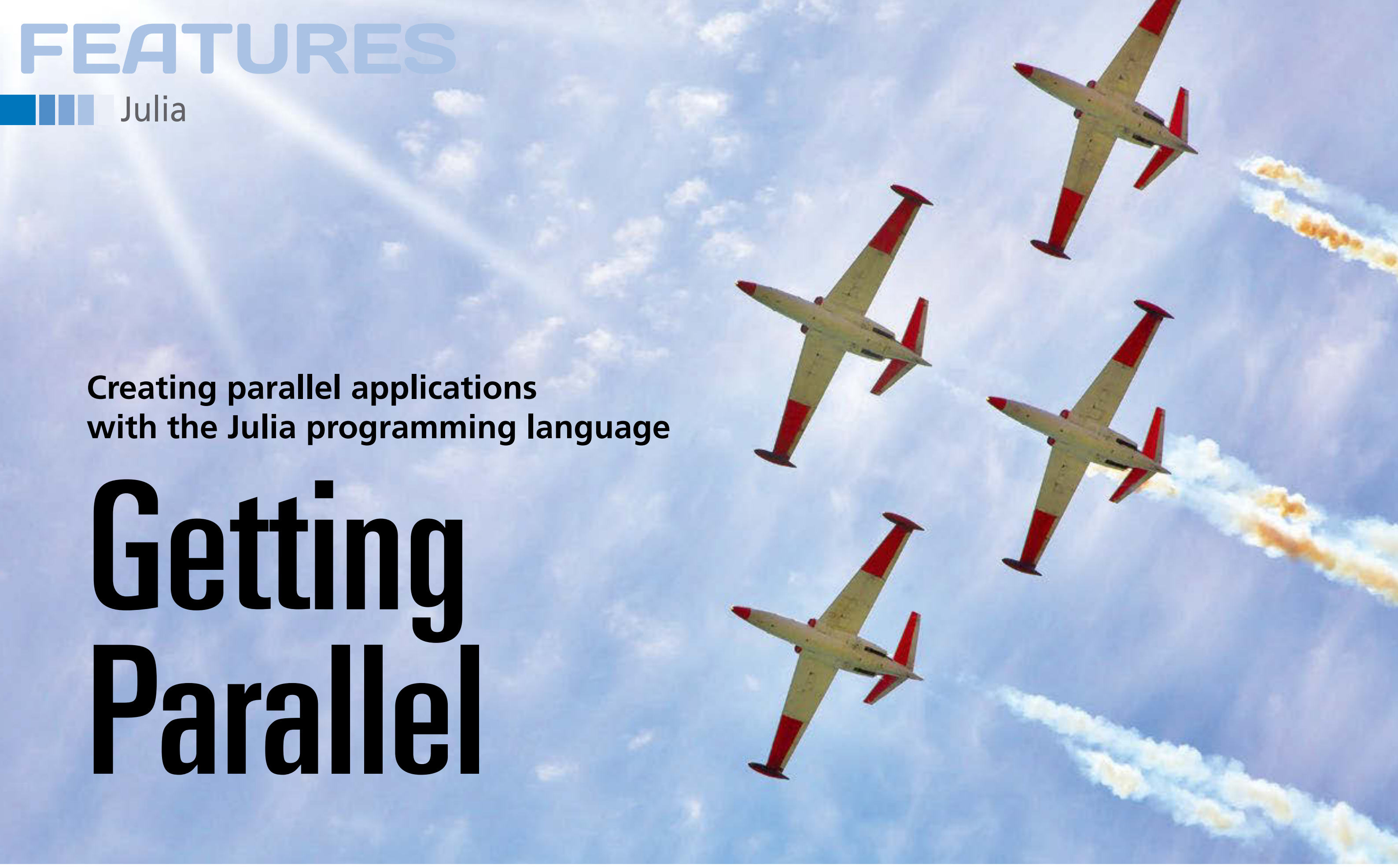
Just remember to run through your "flow control" settings, test TSO both on and off, and force your speed and duplex settings. Fire up a tool like `ngrep` to sniff the data as it traverses your links, and bear in mind the fun and games you can experience with auto-negotiation too.

See the box titled "Mii-tool" for a look at an alternative for older systems that might not support ethtool. ■■■

INFO

[1] Ethtool: *http://www.linuxcommand.org/man_pages/ethtool8.html*

[2] Ethtool Fedora Docs: *http://docs.fedoraproject.org/en-US/Fedora/18/html/System_Administrators_Guide/s2-ethtool.html*



Creating parallel applications with the Julia programming language

Getting Parallel

Parallel processing is indispensable today – particularly in the field of natural sciences and engineering. Normal desktop users, however, can also benefit from higher performance through parallel execution with at least four calculation cores.

By Mark Vogelsberger

Programming tools such as MPI and OpenMP offer parallel processing features. It is easy to use these parallel language extensions, but using them efficiently is difficult because many algorithms cannot be rewritten for them. Languages such as Python and R also include parallel extensions, but these extensions were added on after the original language development and tend to be extremely slow when it comes to numerical calculations.

Many developers are looking for a language that is specifically designed with the intention of supporting parallel processing, and they want this parallel language to be easy to handle, with built-in features that facilitate parallelization and offer performance close to the blazing speed of C. A new language called Julia was developed to fill this niche (see the box titled “Julia Performance”).

Hello Julia!

The current Julia version is available from GitHub [1]:

```
git clone https://github.com/⮐
  JuliaLang/julia.git
```

Most Linux distributions also include Julia packages. Julia is a very young language that is constantly in development, so if you want the latest features, you should look for the latest developer release.

If you get the latest Julia source code from GitHub, you’ll need to compile it using `make`. Compiling the code could take a few minutes – it should all run smoothly on most systems; however, the `README.md` file explains some cases where you need to manually readjust. After you compile, the `julia` binary files will appear in the root directory.

When you start it, you will see the Julia prompt shown in Figure 1. The dance starts when you enter a command at the Julia prompt. Programmers can try a Hello World example:

```
julia> for i=1:10
 println("Hello World ",i)
 end
```

produces the following output:

```
Hello World 1 Hello World 2 ⮐
  Hello World 3 ...
```

Julia’s syntax is heavily based on MATLAB. For example, indexing arrays starts at 1, not at 0. The syntax of for-loops is also the same as MATLAB. See the official Julia documentation [1] for more on the basic language structure.

Parallel with Julia

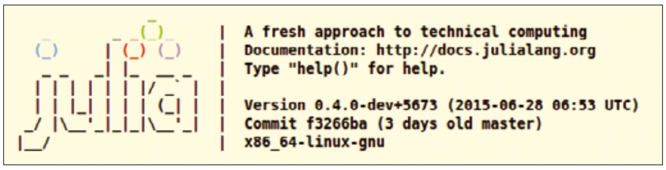
To parallelize loops using threading, programmers break up the loop and assign a part of the loop to each of the system’s processors. The whole loop is processed faster because all the processors are computing in parallel. Another popular form of parallelization is distributed-memory parallelization, which involves calls for distributing both the computing work and the data.

Julia still uses another approach – a master-worker architecture or a one-sided-message passing. A process (the master) gives instructions to other processors (slaves). Programmers therefore only need to explicitly manage a process. The whole Julia parallelization is based

Lead Image © Kavram, 123RF.com

on two main structures: remote references and remote calls.

A remote reference refers to an object that is connected to another process. For example, the result from a computation provided by another process can be used locally by using a reference. Similarly, a remote call is a call that allows a function to be executed by another process. Programmers can let any number of Julia functions be executed via remote call by any number of processes. All other parallel language structures in Julia use remote references and remote calls to deal with parallel tasks.



```
               _
   _       _ _(_)_     |  A fresh approach to technical computing
  (_)     | (_) (_)    |  Documentation: http://docs.julialang.org
   _ _   _| |_  __ _   |  Type "help()" for help.
  | | | | | | |/ _` |  |
  | | |_| | | | (_| |  |  Version 0.4.0-dev+5673 (2015-06-28 06:53 UTC)
 _/ |\__'_|_|_|\__'_|  |  Commit f3266ba (3 days old master)
|__/                   |  x86_64-linux-gnu
```

Figure 1: Julia's welcome screen right after the start.

To operate Julia in parallel, users need to inform the program at the start how many processes should be started. If, for example, the machine has four processor cores, it could start four processes using the following:

```
./julia -p 4
```

You could also start more processes, although this wouldn't see any improvement in performance if you only had four physical cores.

You can add processes on the fly using the `addprocs()` function. Each process has an ID – the first master process is assigned ID 1. All worker processes have IDs higher than 1. The `myid()` function returns a process's ID.

Julia can also run in parallel on whole clusters of computers. A parallel cluster requires a password-free SSH environment. Julia then adds individual cluster computers using the `machinefile` option.

Anyone wanting to have a task executed by a particular process needs to use the `remotecall()` low-level function:

```
julia> result = remotecall(3, +, 2, 2)
RemoteRef(3,1,6)
```

The process with ID 3 calculates 2 + 2, a trivial example. However, the immediate response isn't the result, but rather a remote reference (`RemoteRef`), because the result isn't available locally as process 3 calculated it. To receive the result, you must first collect it from the remote process via `fetch()`:

```
julia> fetch(result)
4
```

As mentioned earlier, `remotecall()` is a low-level function. If it doesn't matter which process executes, you can use Julia's `spawn` macro to send tasks to other processes. The `spawnat` macro also makes it possible to specify the process. For example,

```
julia> @spawnat 2 rand(10,10)
```

creates a random matrix through process 2, and

```
julia> @spawn rand(10,10)
```

JULIA PERFORMANCE

Table 1 shows some of the micro benchmarks for different programming languages and problems published on the Julia website. The values are normalized so that C is equal to 1 and smaller values indicate better performance. Julia has almost C-like performance for some of the micro benchmarks and is much faster than languages like R and Python.

Other examples confirm this performance edge. For instance, you can use either Python or Julia to calculate many square roots in a loop. The examples explicitly write the loop from:

```
import math
import time
tstart = time.time()
for i in xrange(100000000):
 math.sqrt(i)
tstop = time.time()
print tstop-tstart
```

This loops needs about 10 seconds on the test system. The following Julia code executes exactly the same operations:

```
tstart = time()
for i = 1:100000000
 sqrt(i);
end
tstop = time()
println(tstop-tstart)
```

and takes less than 0.1 seconds. The Julia code is therefore more than 100 times faster than the equivalent Python code. Multiple dispatch with function calls gives Julia extremely efficient code that is practically superior to any high-level language. Faster code in Julia can be achieved without any tricks like vectorization or outsourcing to C extensions. By contrast, such tricks are often necessary to speed up Python or R code.

TABLE 1: Micro Benchmarks

	Fortran	Go	Java	JavaScript	Julia	MATLAB	Python	R
Version	Gcc 4.8.2	01.02.01	1.7.0_75	V8 3.14.5.9	0.3.7	R2014a	02.07.09	3.13
Fib	0.57	2.20	0.96	3.68	2.14	4258.12	95.45	528.85
Parse_int	4.67	6.09	5.43	2.29	1.57	1525.88	20.48	54.30
Quicksort	1.10	2.00	1.65	2.91	1.21	55.87	46.70	248.28
Mandel	0.87	0.71	0.68	1.86	0.87	60.09	18.83	58.97
Pi_sum	0.83	1.00	1.00	2.15	1.00	1.28	21.07	14.45
Rand_mat_stat	0.99	3.71	4.01	2.81	1.74	9.82	22.29	16.88
Rand_mat_mul	4.05	1.23	2.35	14.58	1.09	1.12	1.08	1.63

leaves the choice of process to Julia. Depending on the algorithm and problem, programmers can therefore decide whether to perform the process distribution themselves or to leave it totally to Julia.

It is usually easiest to let Julia get on with it. `spawn` also only returns a remote reference, which then again requires a `fetch()` call to collect the result:

```
julia> result = @spawn 2+2
RemoteRef(2,1,6)
julia> fetch(result)
4
```

Calling `remotecall()` or `spawn` doesn't guarantee that the immediately-returned remote reference will also contain the result. Only `fetch()` ensures this. For example, `spawn` could start a complex and lengthy computation using another process. Nevertheless, `spawn` won't initially block anything.

However, `fetch()` can only deliver the result once the calculation is also finished. That means it is `fetch()` that then blocks for a time and the local process waits until the result is available.

Distributed Objects

Julia also offers the possibility to directly generate distributed objects. This feature is particularly helpful when working with large matrices. Julia makes available distributed arrays that are of type `Darray`. To use these arrays, users first need to install the necessary package through the internal package management (`Pkg`):

```
julia> Pkg.checkout↩
  ("DistributedArrays")
```

This command downloads and installs the `DistributedArrays` code. You can get an overview of all installed packages using `Pkg.status()`. If updates for packages are available, you can easily install them using `Pkg.Update()`. To use distributed arrays, you need to ensure that all processes load the package:

```
julia> @everywhere using ↩
  DistributedArrays
```

The `everywhere` macro loads the package via `using` and ensures that it is available to all processes.

Julia offers numerous ways to generate distributed arrays. The easiest way is using the following functions:

```
dzeros(100,100,10)
dones(100,100,10)
drand(100,100,10)
drandn(100,100,10)
dfill(x,100,100,10)
```

These functions provide distributed arrays with the specified dimensions and characteristics. For example, `dzeros` provides an array filled with zeros. Julia sorts all this work out: The individual matrix elements are then distributed to the various processes and initialized.

Julia provides other useful functions for working with distributed arrays. The `localpart()` function, for example, returns part of the array that is assigned to the local process. `localindexes()` analogously returns the indices of the local part of the array.

If the local process is supposed to edit a distributed array, programmers can use `convert()`. `distribute()` converts a local array into a distributed array (i.e., the function distributes the array's data to the existing processes).

Initialization

In addition to the functions for creating arrays, Julia supports complicated initialization functions, such as:

```
DArray(init, dims[, procs, dist])
```

Programmers need to supply the `init` function, which is given a tuple of index ranges for which the array is then initialized. `dims` specifies the array's dimensions.

Programmers can use `dist` to determine how the array will be distributed across the various processes. However, this manual declaration isn't generally necessary because Julia sorts it automatically. The example in Listing 1 creates a distributed random matrix and then returns the part for which Process 2 is responsible.

The practical thing about these distributed arrays is that Julia hides all communications from the user. If, for example, a user wants to calculate the sum of all the elements of the `d` matrix, that user just needs to call `sum(d)`. `sum(d)` delivers the result directly. Julia adds all the elements of the subarray with each individual process. All partial results to the total are then added.

Julia also offers structures for easily distributing loops. The following few lines, for example, add generated random numbers using `rand()`:

```
julia> r = @parallel (+) ↩
  for i=1:200000000
 rand()
 end
```

The `@parallel` macro distributes the loop to all the available processes. The return value for each loop is the last expression in the loop, in this case, the `rand()` statement for generating a random number. The example above then adds the results for each loop run. Programmers can also omit the reduction operator (in this case, the addition). Julia then distributes the loop in parallel without reducing the results at the end.

The `pmap()` function is useful for such cases. `pmap()` simply executes a function for a specific object (i.e., a typical map

LISTING 1: Distributed Arrays

```
julia> @everywhere using DistributedArrays

julia> d=drand(10,10);

julia> r=@spawnat 2 begin
 localpart(d)
 end

RemoteRef(2,1,23)

julia> fetch(r)
5x5 Array{Float64,2}:
 0.932366 0.181326 0.0261475 0.211363 0.308226
 0.330008 0.924271 0.543386 0.895825 0.617452
 0.51471 0.801718 0.786854 0.174585 0.413264
 0.840219 0.750775 0.126154 0.853618 0.899762
 0.866529 0.804654 0.19808 0.49411 0.951355
```

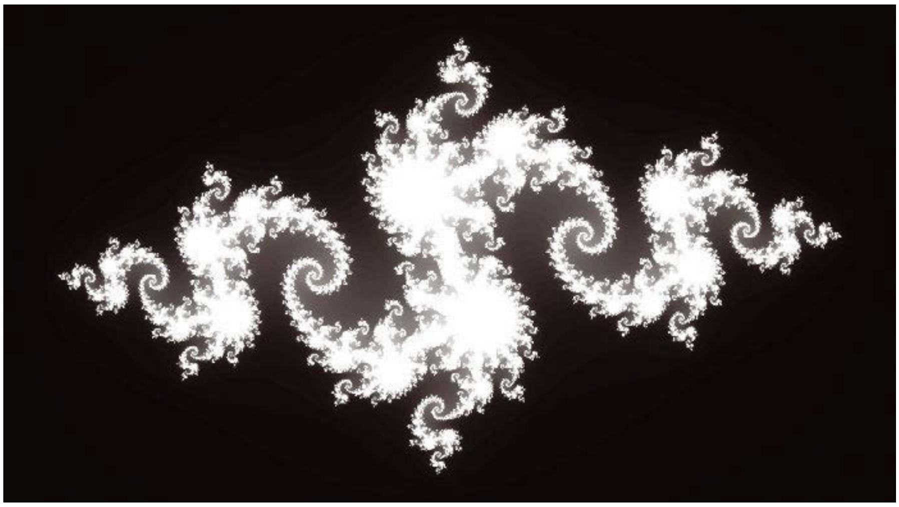


Figure 2: Julia calculates Julia sets with distributed arrays.

task). The following example uses the `pmap()` statement to compute the rank of matrices:

```
julia> M = ↩
  [rand(1000,1000) for i=1:4];
julia> pmap(Rang,M)
4-element Array{Any,1}:
 1000
 1000
 1000
 1000
```

The first statement here creates four 1000x1000 matrices. `pmap()` then calculates the rank for each matrix. Because Julia is started with `-p 4`, a worker process is responsible for each matrix. It is just as easy to calculate the characteristics of other matrices. For example, the `det()` function calculates the determinant, and the `inv()` function calculates the inverse of a matrix.

Monte Carlo Pi Calculation

Monte Carlo calculations are generally very easy to parallelize because the individual calculation steps are mostly independent of each other.

LISTING 2: Monte Carlo Estimation

```
N_tot = 1000000000

tstart = time()

N_in = @parallel (+) for n=1:N_tot
        x = rand() * 2 - 1
        y = rand() * 2 - 1

        r2 = x*x + y*y
        if r2 < 1.0
                1
        else
                0
        end
end

tstop = time()

pi_MC = N_in/N_tot*4.0

println("time        = ", tstop-tstart, " seconds")
println("pi estimate = ", pi_MC)
```

For Monte Carlo-based calculations with the number pi, programmers need to generate a lot of random numbers in a square surrounding the unit circle (radius 1). These random numbers are generated uniformly between -1 and +1 for the x- and y-coordinates.

The ratio of the area of the unit circle to the square is now just pi*1*1/(2*2). This means that exactly the proportion pi/4 of the random numbers should be in the circle. If you now

LISTING 3: Julia Set

```
#DistributedArrays, WIDTH, HEIGHT and MAXITER are global
@everywhere using DistributedArrays
@everywhere WIDTH=1000
@everywhere HEIGHT=500
@everywhere MAXITER=100

#Images is local
using Images



# Julia function
@everywhere function julia(z, maxiter::Int64)
 c=-0.8+0.156im
 for n = 1:maxiter
 if abs(z) > 2.0
 return n-1
 end
 z = z^2 + c
 end
 return maxiter
end



# Init function for creating the array
@everywhere function parJuliaInit(I)
 # local patch
 d=(size(I[1], 1), size(I[2], 1))
 m = Array(Int, d)

 xmin=I[2][1]
 ymin=I[1][1]

 for x=I[2], y=I[1]
 c = complex((x-WIDTH/2)/(HEIGHT/2), (y-HEIGHT/2)/
            (HEIGHT/2))
 m[y-ymin+1, x-xmin+1] = julia(c, MAXITER)
 end
 return m
end



# creates distributed array
Dm = DArray(parJuliaInit, (HEIGHT, WIDTH))

# fetches distributed array on the local process
m = convert(Array, Dm)

saves #image as PNG
imwrite(grayim(transpose(m)/(1.0*MAXITER)),"juliaset.png")
```

count them in a loop, you will get the Monte Carlo estimate for pi. In the following example:

```
pi_MC = N_in / N_tot * 4
```

`N_in` is the number of random numbers in the unit circle, and `N_tot` the total number of random numbers generated in the x-y plane. The higher `N_tot`, the closer `pi_MC` is to pi. The best thing is, therefore, to generate loads of random numbers to obtain a very precise value for pi.

Listing 2 shows an implementation of this method in Julia. The program executes the central loop using `@parallel` and adds the loop result (1 or 0) at the end in a reduction step. This code runs on a processor in less than two minutes on the test system:

```
time = 116.90136814117432 seconds
pi estimate = 3.14162308
```

Anyone who uses two cores will speed up the example by nearly a factor of 2:

```
time = 73.63914084434509 seconds
pi estimate = 3.141607444
```

The method converges very slowly, but the first two digits of pi (3.14) have already been calculated correctly.

A Second Example: Julia with Julia

The parallel calculation of a Julia fractal with Julia is a more complicated example. Martin Rupp originally wrote the program based on the official Julia Mandelbrot example [2]. However, the version shown in Listing 3 is modified because the syntax has changed.

The mathematical idea of a Julia fractal is very simple. Consider the sequence of complex numbers $zn + 1 = zn + c$ for an arbitrary complex constant c. Each c yields a particular Julia set. Typical Julia images are created by using a counter. The more iterations are needed in the Zn-series to reach a certain threshold value, the brighter the associated item appears. The two-dimensional complex plane can be easily mapped to the x- and y-coordinate of a pixel grid through which the known pictures are created.

Julia can easily deal with complex numbers, which is why the implementation is particularly simple. The code is complicated by parallelization because distributed arrays are used. Depending on the process ID, part of the Julia image is calculated by another process. An advantage of parallelization is that the calculation of a certain pixel is independent of its surroundings. The situation would be more complicated if the characteristic of one pixel also depended on the environment of the pixel. So, using

```
~/julia/julia -p 4 juliaset.jl
```

it is possible to easily execute code (see Figure 2).

Conclusions

Although Julia is still a young language, you can already use it to write plenty of productive code. Julia is practically unbeatable – especially for applications in the field of numerical calculations – and it will someday replace older toolsets like Python, R, or MATLAB. ■■■

INFO

[1] Julia Documentation: *http://docs.julialang.org/en/release-0.3/*

[2] Original Julia sets example: *http://mathemartician.blogspot.de/2012/07/julia-set-in-julia.html*

Smart research using Elasticsearch

More, Please

Websites often offer readers links to articles about similar topics. Using Elasticsearch, the free search engine, is one way to find related documents instantly and automatically. By Mike Schilli

When people rummage around on the StackOverflow website looking for advice on programming questions, they can find list of links to related topics in the Related section (Figure 1). This helps users if the first search result didn't show what they expected or the located resource is insufficient. According to Gormley and Tong [1], Elasticsearch [2] [3], the free search engine, generates these links on the website in real time from the growing and very impressive collection of 10 million StackOverflow contributions.

Artificial Brain

This isn't actual intelligence at work, because computers still find it difficult to understand the content of a document, meaning they can't find documents with related content. In fact, the algorithm used is based on simple nitpicking – it combines values for word frequency and derives a score from those values.

Elasticsearch uses an inverted index for this task; this index is a complete list of individual words that appear in any of the documents that have been added to the search engine so far. It remembers which document each word has been found in and can therefore instantly output a list of documents for a search term.

For example, if a user is looking for the term *perl*, Elasticsearch will immediately find the doc-1 document in the inverted index from Figure 2 and present this (one hopes) accurate search result.

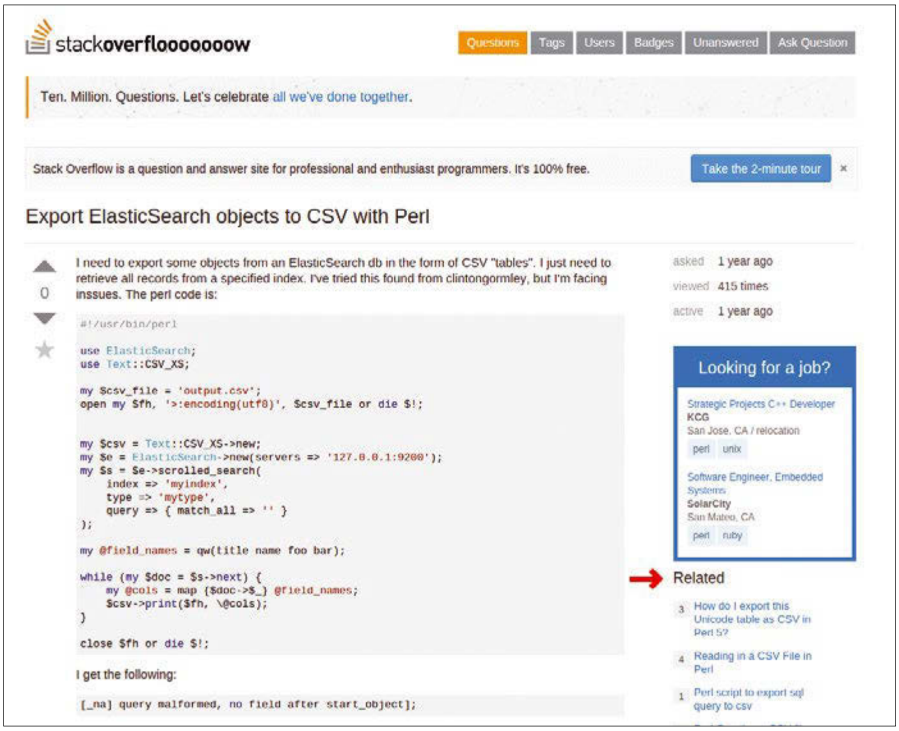
When searching for two words (e.g., *linux* and *cpan*), two documents come into consideration, but because doc-2 only contains one term, whereas doc-3 contains both, the algorithm gives doc-3 a higher relevance score. In a match list sorted by descending score, doc-3 is then at the very top and is more likely to match the user's expectations.

What Is Relevant?

Not all words are equally important. For example, the word *file* understandably comes up in quite a large number of documents on computer topics. If the user searches for *file linux cpan*, doc-2 and doc-3 provide two matches each, but be-

MIKE SCHILLI

Mike Schilli works as a software engineer in the San Francisco Bay Area. He can be contacted at *mschilli@perlmeister.com*. Mike's homepage can be found at *http://perlmeister.com*.



stackoverflooooooow

Ten. Million. Questions. Let's celebrate all we've done together.

Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free.

Export ElasticSearch objects to CSV with Perl

I need to export some objects from an ElasticSearch db in the form of CSV "tables". I just need to retrieve all records from a specified index. I've tried this found from clintongormley, but I'm facing inssues. The perl code is:

```
#!/usr/bin/perl

use ElasticSearch;
use Text::CSV_XS;

my $csv_file = 'output.csv';
open my $fh, '>:encoding(utf8)', $csv_file or die $!;


my $csv = Text::CSV_XS->new;
my $e = ElasticSearch->new(servers => '127.0.0.1:9200');
my $s = $e->scrolled_search(
    index => 'myindex',
    type => 'mytype',
    query => { match_all => '' }
);

my @field_names = qw(title name foo bar);

while (my $doc = $s->next) {
    my @cols = map {$doc->$_} @field_names;
    $csv->print($fh, \@cols);
}

close $fh or die $!;
```

I get the following:

```
[_na] query malformed, no field after start_object];
```

asked 1 year ago
viewed 415 times
active 1 year ago

Looking for a job?

Strategic Projects C++ Developer
KCG
San Jose, CA / relocation
perl unix

Software Engineer, Embedded Systems
SolarCity
San Mateo, CA
perl ruby

Related

3 How do I export this Unicode table as CSV in Perl 5?

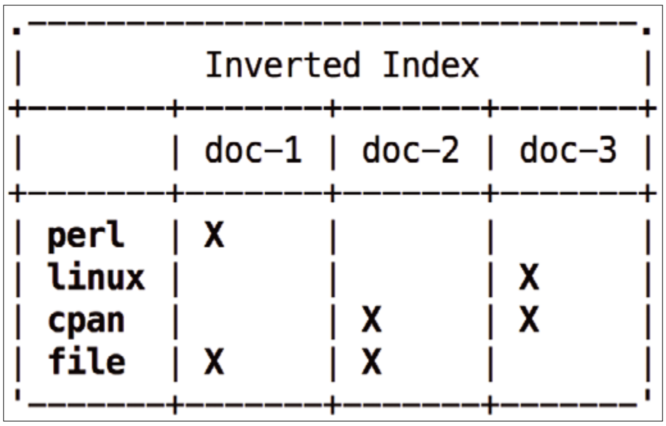
4 Reading in a CSV File in Perl

1 Perl script to export sql query to csv

Figure 1: Using Elasticsearch, StackOverflow generates links to similar topics in the Related section.

cause *linux* is more significant for one document than *file*, the algorithm rates *linux cpan* higher than *cpan file* and gives doc-3 preference.

The Tf-idf score [4] determines how important a word is within a document. It gives a high value to those words that are prevalent in one document but do not occur too frequently in other documents competing for a high score (i.e., words that underpin the uniqueness of the document). A word's relevance value increases with the number of times the word appears in the document (this is known as term frequency, TF) and decreases if the word also appears in many other documents in the collection (IDF, inverse document frequency).



```
.---------------------------------------.
|             Inverted Index            |
+---------+---------+---------+---------+
|         | doc-1   | doc-2   | doc-3   |
+---------+---------+---------+---------+
| perl    | X       |         |         |
| linux   |         |         | X       |
| cpan    |         | X       | X       |
| file    | X       | X       |         |
'---------+---------+---------+---------'
```

Figure 2: The inverted index maintains a list of documents in which the indexer found specific words.

Searching for the Same

To find documents in the database that are similar to document *x*, Elasticsearch first extracts all relevant words from *x*, then forms a search query using these words and returns the results. Elasticsearch performs this search for similar documents using the `more_like_this` query [5] command with very little programming required. However, all the relevant documents must be added to the index beforehand. I'll be using the official Perl client Search::Elasticsearch from CPAN for this.

Listing 3 (described later) wades through a directory of text files. These are stories from my *Usarundbrief.com* blog that I extracted from the home-grown content management system via another Perl script. Each text file corresponds to a blog entry – 877 messages accumulated over almost 20 years, which is why the directory contains 877 files. Listing 1 shows the shell output [6].

Structure of the Index

Invoking `mlt-index` (Listing 3) from the command line grabs the files one by one and feeds them via the `index()` method to the Elasticsearch server installed on my desktop computer. (See "Installing the Elasticsearch Server" box.)

If you would rather not clutter your filesystem with yet more software, you can simply boot a VM using `vagrant up` with the Vagrant file shown in Listing 2, and install the server in the VM. The `forwarded_port` mapping ensures that the Elasticsearch server in the VM listing on port 9200 is also responsive on the host system on port 9200.

Looking at Listing 3, I simply called the newly created index in the server `blog`. Line 15 deletes it first of all if it is already there – which is typically after previously calling `mlt-index`.

The `find()` function in line 17, originating from the File::Find module accompanying Perl, then recursively works

LISTING 1: Feed Data

```
$ ls idx | wc -l
877
$ mlt-index
Added 10-cents-for-a-grocery-bag.txt
Added a-job-for-angelika.txt
Added absurd-and-funny-american-tv-shows.txt
...
```

LISTING 2: Vagrant File

```
VAGRANTFILE_API_VERSION = "2"

Vagrant::configure(VAGRANTFILE_API_VERSION) do |config|

      # 32-bit Ubuntu Box
    config.vm.box = "ubuntu/trusty64"
    config.vm.network "forwarded_port", guest: 9200, host: 9200
end
```

through all the files under the specified directory and, behind the scenes, switches to the directory in which they're located during the user-defined callback using `chdir`. It also sets the name of the file currently being edited in the variable `$_`.

Feeding by Slurping

The `index()` method in line 23 adds the file's text content slurped with `slurp()` from the Sysadm::Install CPAN module's treasure trove, along with the file name to the search engine's index. Later, the search functions return the file names with matches, and they are easy to recognize because the names were chosen according to the content of each blog entry (e.g., `10-cents-for-a-grocery-bag.txt`).

The `index()` feed function also creates a new index (if it doesn't already exist) and defines both the item's name (`blog`) and a `type` (set to `text`) – which in Elasticsearch is nothing more than an arbitrarily named partition in the index.

More of That!

Feeding data to Elasticsearch took about a tenth of a second per 2KB text file on my Linux desktop; this therefore really tested my patience with the 877 blog entries. By contrast, feeding data on my laptop with a faster solid state disk and plenty of memory was much faster – it was all over within 10 seconds.

Queries can be submitted as soon as the index is ready. The output from the `mlt-search` command in Figure 3 shows that Listing 4 unsurprisingly rediscovers the document I provided as a reference – content related to my family dealing earthquakes in the San Francisco Bay area. But, it also dug up some other earth-shaking results, such as a report about how to traverse the bureaucratic jungle to obtain a driver's li-

INSTALLING THE ELASTICSEARCH SERVER

The Elasticsearch server can be easily installed on a desktop computer or a virtual machine using the following simple steps:

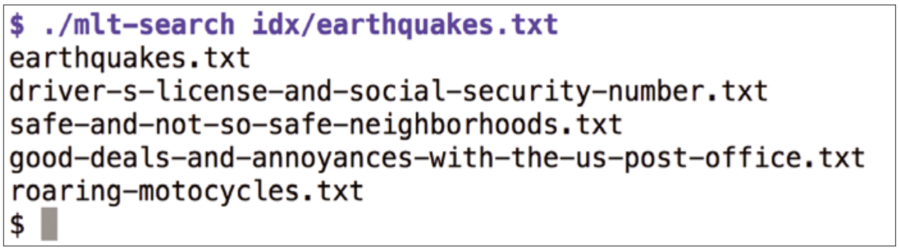
```
sudo add-apt-repository ppa:webupd8team/java
sudo apt-get update
sudo apt-get install oracle-java7-installerjava -version
wget https://download.elastic.co/elasticsearch/↩
    elasticsearch/elasticsearch-1.7.1.deb
sudo dpkg -i *.deb
sudo /etc/init.d/elasticsearch start
```

LISTING 3: mlt-index

```
01 #!/usr/local/bin/perl -w
02 use strict;
03 use Search::Elasticsearch;
04 use File::Find;
05 use Sysadm::Install qw( slurp );
06 use Cwd;
07 use File::Basename;
08
09 my $idx = "blog";
10 my $base_dir = getcwd;
11 my $base = $base_dir . "/idx";
12
13 my $es = Search::Elasticsearch->new( );
14 eval {
15   $es->indices->delete( index => $idx ) };
16
17 find sub {
18   my $file = $_;
19
20   return if ! -f $file;
21   my $content = slurp $file, { utf8 => 1 };
22
23   $es->index(
24     index => $idx,
25     type  => 'text',
26     body  => {
27       content => $content,
28       file    => $file,
29     }
30   );
31   print "Added $file\n";
32 }, $base;
```

LISTING 4: mlt-search

```
#!/usr/local/bin/perl -w
use strict;
use Search::Elasticsearch;
use Sysadm::Install qw( slurp ) ;

my $idx = "blog";

my( $doc ) = @ARGV;
die "usage: $0 doc" if !defined $doc;

my $es = Search::Elasticsearch->new( );

my $results = $es->search(
  index => $idx,
  body  => {
    query => {
      more_like_this => {
        like_text =>
          slurp( $doc, { utf8 => 1 } ),
        min_term_freq   => 5,
        max_query_terms => 20,
      }
    }
  }
);

for my $result (
  @{ $results->{ hits }->{ hits } } ) {

  print $result->{ _source }->{ file },
    "\n";
}
```



```
$ ./mlt-search idx/earthquakes.txt
earthquakes.txt
driver-s-license-and-social-security-number.txt
safe-and-not-so-safe-neighborhoods.txt
good-deals-and-annoyances-with-the-us-post-office.txt
roaring-motocycles.txt
$ 
```

Figure 3: Elasticsearch found similar articles for a blog entry about earthquakes in the San Francisco Bay area.

cense in California, about how to distinguish good and bad neighborhoods to live in, and the annoying habit of some motorcycle riders to let their engines roar at earth-shattering levels. The results appear in fractions of a second, meaning the function is certainly also useful on busy websites.

Listing 4 expects a text file's path at the command line and imports this file using `slurp`, while telling Perl to keep the `utf8` encoding intact. It sends the query to the Elasticsearch server in line 13 using the `search()` method and then collects the names of the files from the matches with comparable content from the result returned as JSON.

Fine-Tuning

The `min_term_freq` parameter specifies a threshold for the selection of a word in the reference document with the `more_like_this` function. If `min_term_freq` is set to the default value `2`, a word must occur there at least twice to make its way into the list of words with which other documents are compared later. The second parameter `max_query_terms` is the maximum number of words from the list in the original document that the algorithm selects to use later in the query.

For anyone wanting to find out about other methods for fine-tuning the search engine, I would recommend the O'Reilly book on the topic [1]. It explains how to deal with Elasticsearch using examples, provides tips for scaling in clusters, and takes a look behind the scenes, where the Apache Lucene search engine is at work. ■■■

INFO

[1] Gormley, Clinton and Zachary Tong, *Elasticsearch: The Definitive Guide*: O'Reilly, 2015.

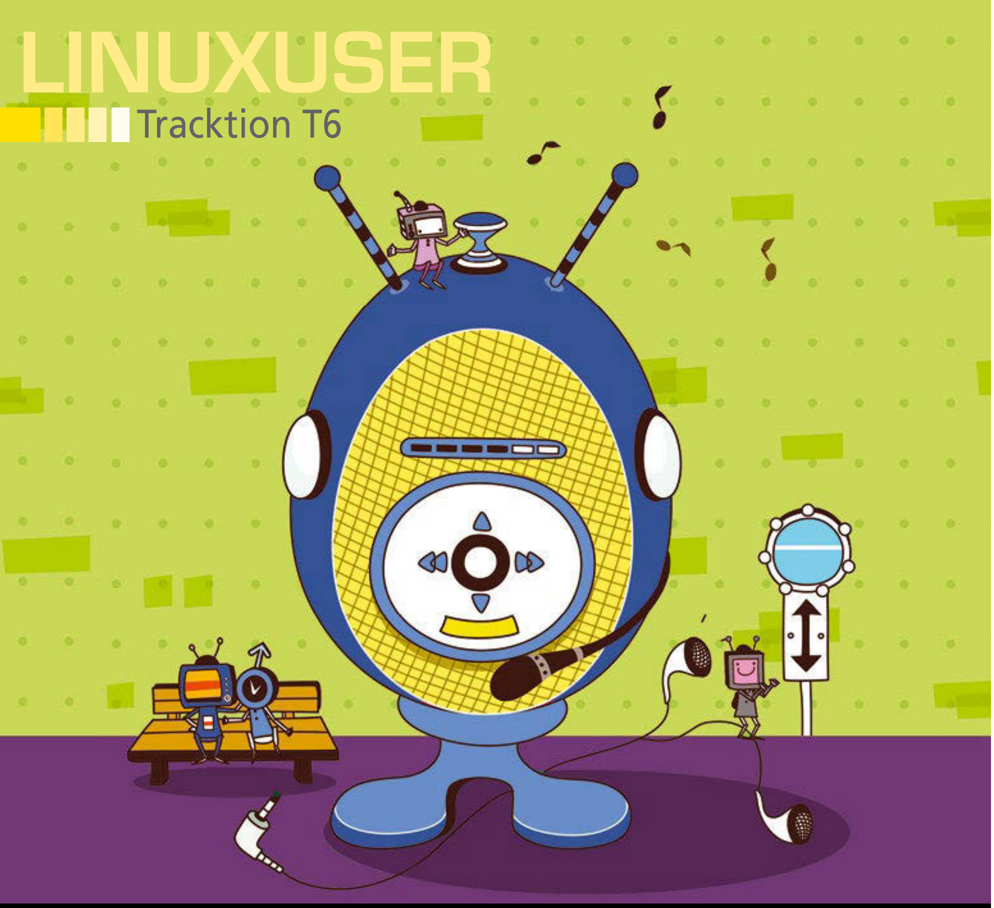
[2] Elasticsearch: *https://www.elastic.co*

[3] "Perl: Elasticsearch" by Mike Schilli, *Linux Magazine*, issue 162, pg. 66, 2014: *http://www.linux-magazine.com/Issues/2014/162/Perl-Elasticsearch/(language)/eng-US*

[4] Tf-idf: *https://en.wikipedia.org/wiki/Tf%E2%80%93idf*

[5] More Like This Query: *https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-mlt-query.html*

[6] Listings for this article: *ftp://ftp.linux-magazine.com/pub/listings/magazine/182/*



Digital audio workstation Tracktion T6 at a glance

Fast and Fluid

The Tracktion digital audio workstation is finally taking off as it reaches version 6. *By Hartmut Noack*

Musicians have a growing number of options in the Linux ecosystem. Some well-established digital audio workstation (DAW) suites, such as Ardour or Qtractor – and several hundred effects and instruments – are available under a free license. However, proprietary software is also increasingly making its way into the market. One example is Tracktion T6, a low-budget DAW. Version 6 of Tracktion has been officially available for Linux since the summer (Figure 1).

Just like Home

The Tracktion Software Company [1] has offered the program for Linux since 2013. The software was essentially designed as a Windows program, but uses Juce (see "The Juce Library" box) for its audio engine and interface. Juice author Julian Storer is active in the Linux audio scene and has published a wide range of plugins and tools for Linux musicians under free licenses.

Tracktion T6, like its predecessors, is available as a Debian package built in Ubuntu. However, the software doesn't explicitly require a specific version of Ubuntu and also works with other Debian derivatives. You can also convert the package into an RPM archive, which you can install in Fedora and openSUSE using Alien.

Soon after its initial launch, the software displays better integration with Linux than any of its predecessors: Both the interface and the internal functions easily mesh with the usual systems in Linux. Some small problems that arose in Tracktion 4 and 5 have been eliminated. For example, Tracktion T6 connects to the Internet with no problem for downloading language files. Users can now also unlock registered licenses without any issues.

Searching for new plugins – which used to be quite tricky – is now easy. The application used to freeze with incompatible modules; it now just skips such files and displays them in a window as a reference at the end of the scan (Figure 2).

THE JUCE LIBRARY

Unlike cross-platform programs like Bitwig Studio or Minecraft, Tracktion does not use Java but rather Juce [2], which developer Julian Storer created himself. The software is freely licensed. The core components are licensed under ISC, the extensions under GPL. After downloading the ZIP archive from the website [3], you can set up projects using a wizard.

As well as various categories for editing audio data, Juce also offers elements for a program interface and for processing strings and image formats. Juce's native programming language is C++; from the outset, it was specifically optimized for high-speed and efficient applications. The very quick reaction times in Tracktion are proof that this concept works (see "Quick Change" box).

Storer has written a whole series of freely available plugins and host programs for musicians in Juce for Linux or ported them from Windows VST plugins. Names like Presonus, Korg, and MAudio can be found among the framework's commercial users. Juce is also used very frequently with mobile applications and games.

AUTHOR

Hartmut Noack (*http://lapoc.de*) works as a lecturer, writer, and musician in Celle and Hanover, Germany. He has always thought that free software and homemade music fit together brilliantly. When he is not sitting in front of his Linux audio workstation, you will find him hanging about on web servers. You can download some CC-licensed musical samples of his work with free music software from Hartmut's own server.

Tracktion T6 cannot use LV2 plugins directly, however, the Linux VST version of the Carla plugin host can be used for this. Only the graphical interfaces for the modules loaded in Carla are problematic. Things work well if you use the tools that are installed generically in Carla for setting up plugins. This also applies to processing audio data, with which LV2 plugins integrated in Carla behave just as stably and flawlessly as other Linux VST effects or instruments (Figure 3).

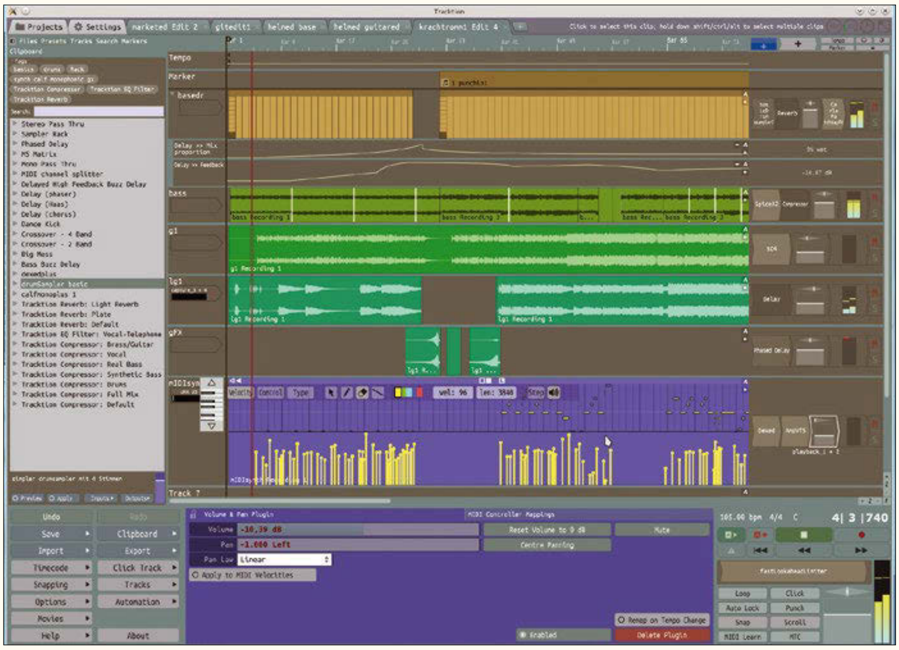
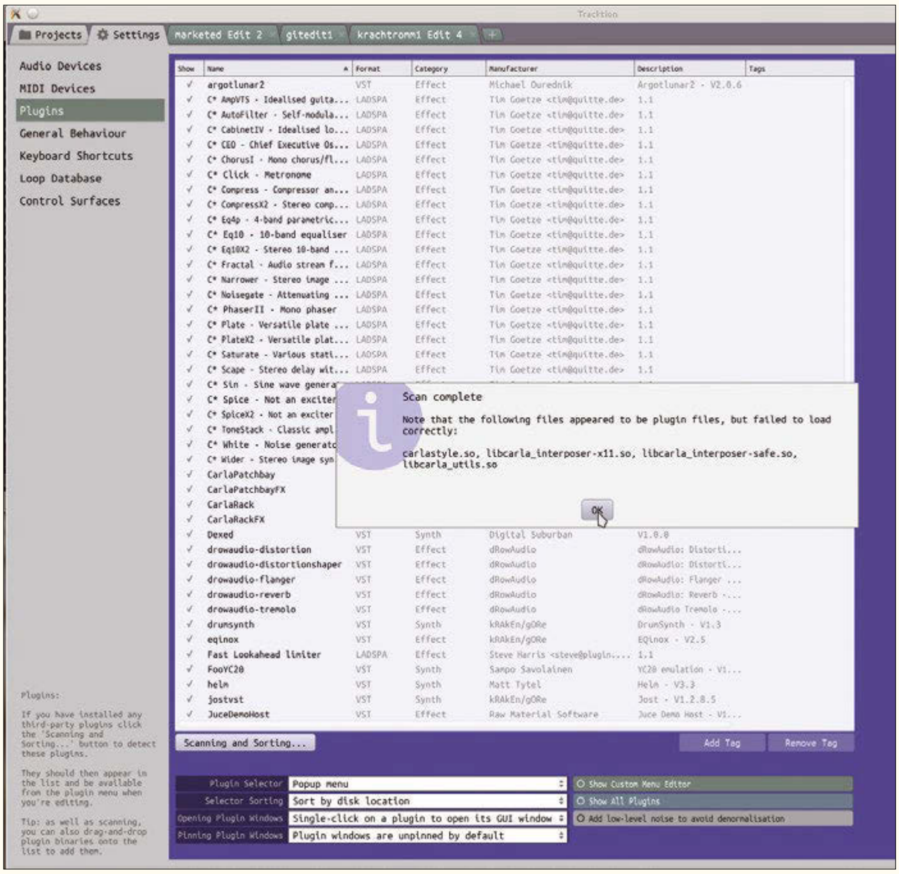


Figure 1: New tracks for automation, new step-sequencer clips, new MIDI functions, and improved system integration make Tracktion 6 an interesting product for musicians using Linux.



Show	Name	Format	Category	Manufacturer	Description	Tags
✓	argotlunar2	VST	Effect	Michael Ourednik	Argotlunar2 - V2.0.6	
✓	C* AmpVTS - Idealised guita...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* AutoFilter - Self-modula...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* CabinetIV - Idealised lo...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* CEO - Chief Executive Os...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* ChorusI - Mono chorus/fl...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Click - Metronome	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Compress - Compressor an...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* CompressX2 - Stereo comp...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Eq4p - 4-band parametric...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Eq10 - 10-band equaliser	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Eq10X2 - Stereo 10-band ...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Fractal - Audio stream f...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Narrower - Stereo image ...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Noisegate - Attenuating ...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* PhaserII - Mono phaser	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Plate - Versatile plate ...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* PlateX2 - Versatile plat...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Saturate - Various stati...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Scape - Stereo delay wit...	LADSPA	Effect	Tim Goetze <tim@quitte.de>	1.1	
✓	C* Sin - Sine wave genera					
✓	C* Spice - Not an exciter					
✓	C* SpiceX2 - Not an exciter					
✓	C* ToneStack - Classic ampl					
✓	C* White - Noise generato					
✓	C* Wider - Stereo image syn					
✓	CarlaPatchbay					
✓	CarlaPatchbayFX					
✓	CarlaRack					
✓	CarlaRackFX					
✓	Dexed	VST	Synth	Digital Suburban	V1.0.0	
✓	drowaudio-distortion	VST	Effect	dRowAudio	dRowAudio: Distorti...	
✓	drowaudio-distortionshaper	VST	Effect	dRowAudio	dRowAudio: Distorti...	
✓	drowaudio-flanger	VST	Effect	dRowAudio	dRowAudio: Flanger ...	
✓	drowaudio-reverb	VST	Effect	dRowAudio	dRowAudio: Reverb -...	
✓	drowaudio-tremolo	VST	Effect	dRowAudio	dRowAudio Tremolo -...	
✓	drumsynth	VST	Synth	kRAkEn/gORe	DrumSynth - V1.3	
✓	eqinox	VST	Effect	kRAkEn/gORe	EQinox - V2.5	
✓	Fast Lookahead limiter	LADSPA	Effect	Steve Harris <steve@plugin....	1.1	
✓	FooYC20	VST	Synth	Sampo Savolainen	YC20 emulation - V1...	
✓	helm	VST	Synth	Matt Tytel	Helm - V3.3	
✓	jostvst	VST	Synth	kRAkEn/gORe	Jost - V1.2.8.5	
✓	JuceDemoHost	VST	Effect	Raw Material Software	Juce Demo Host - V1...	

Scan complete

Note that the following files appeared to be plugin files, but failed to load correctly:

carlastyle.so, libcarla_interposer-x11.so, libcarla_interposer-safe.so, libcarla_utils.so

Plugins:

If you have installed any third-party plugins click the 'Scanning and Sorting...' button to detect these plugins.

They should then appear in the list and be available from the plugin menu when you're editing.

Tip: as well as scanning, you can also drag-and-drop plugin binaries onto the list to add them.

Plugin Selector: Popup menu
Selector Sorting: Sort by disk location
Opening Plugin Windows: Single-click on a plugin to open its GUI window
Pinning Plugin Windows: Plugin windows are unpinned by default

○ Show Custom Menu Editor
○ Show All Plugins
○ Add low-level noise to avoid denormalisation

Figure 2: Some incompatible files in the plugin directory would have caused a crash in the past; however, scanning with the current version of Tracktion is a stress-free process.

QUICK CHANGE

Julian Storer used an unconventional approach for the Tracktion interface from the outset. Everything happens in a window that constantly adapts to the current situation. That this concept no longer sounds so spectacular can be traced to the fact that other software designers have taken the same path after finding out that this approach works well for users of Tracktion.

It is typical for one mouse click (or key combination) to trigger multiple actions on the interface at once. For example, clicking a clip both shows the configuration tool and closes the open plugin interfaces. This is a bit awkward if you still want to do something with the plugin after editing the clip. However, hardly any other programs can open complex interfaces as quickly and smoothly as Tracktion – even though the program practically recomposes the whole interface after some actions.

Users can influence the rebuilding of the interface: The program provides tags that you can use for sound tracks, among other things. Clicking an entry in the list of tags on the left border causes the software to display only the tracks with the corresponding tag. All other tracks in the project continue to run in the background.

PROBLEMS AND SOLUTIONS

When cloning a track with a MIDI sound module by copying and pasting, the sound generator sometimes blocks both tracks. You can solve this problem by creating a new preset for the cloned sound module, then removing the plugin and reinserting it. The previously created preset saves you the arduous process of setting up the desired parameters again.

Not all key combinations work out of the box. Checking the *Settings | Keyboard Shortcuts* will help if an action using the combination suggested in the menu doesn't work. Shortcuts set up in this list worked as expected in the text – curiously, even when the preset combination was simply allocated again.

Carla is able to load plugins in DLL format from Microsoft via a Wine bridge. Although this went well in the test, full functionality was not yet supported. For example, the basic slide controls for plugins installed in Carla (volume, mix), with curves for automation, were usable without any problem, but not the special parameters.

Integrating Tracktion T6 into the world of Jack and Alsa, including the MIDI functions, was much more successful. The MIDI keyboard connected to the PC via USB, and a drum set wired to the MIDI port for the MAudio sound card worked as well as expected with bona fide Linux software such as Ardour. You can combine plugins – if needed – using plugin racks. You can insert racks just like any other plugin by dragging and dropping from the preset list on the left or using the plus arrow at top right.

Tracktion T6 doesn't have any problems with any of these tasks, such as slight delays here or a pop there. Even if the program still announces *Initializing Windows* when launched in the terminal, it obviously feels at home on Linux, except for a few tiny details (see the "Problems and Solutions" box).

Tracktion provides a small, but excellent gift from the world of Windows to Linux. The time-stretch library élastique Pro from zplane [4] is used not only by Tracktion but also in other products, such as Abelton Live and some products from Avid. Stretching and compressing in élastique has made significant progress, regarding both quality and speed.

The *Marketplace* tab is the only new feature that I did not find in Version 6.1.10 (64-bit) for Linux. This tab lets you buy, download, and install extensions and plugins directly from the manufacturer's server and those of some other vendors. However, a look at the Marketplace in a browser [5] shows that nearly all the modules there still only support Mac OS X and Windows.

Musical Scriptorium

Tracktion sees itself as a sequencer: Using Tracktion, you can arrange musical events sequentially on multiple synchronized axes running in parallel and then play these events. The editor provides an array of tracks and a time frame. Although the grid is set to seconds, you can change it to the musical *Bars/Beats* in the *Timecode* menu at the bottom left. If *Snap* is activated at the bottom right, clips and notes in the editor skip to the next line in the grid. Unlike many other DAWs, the program takes the

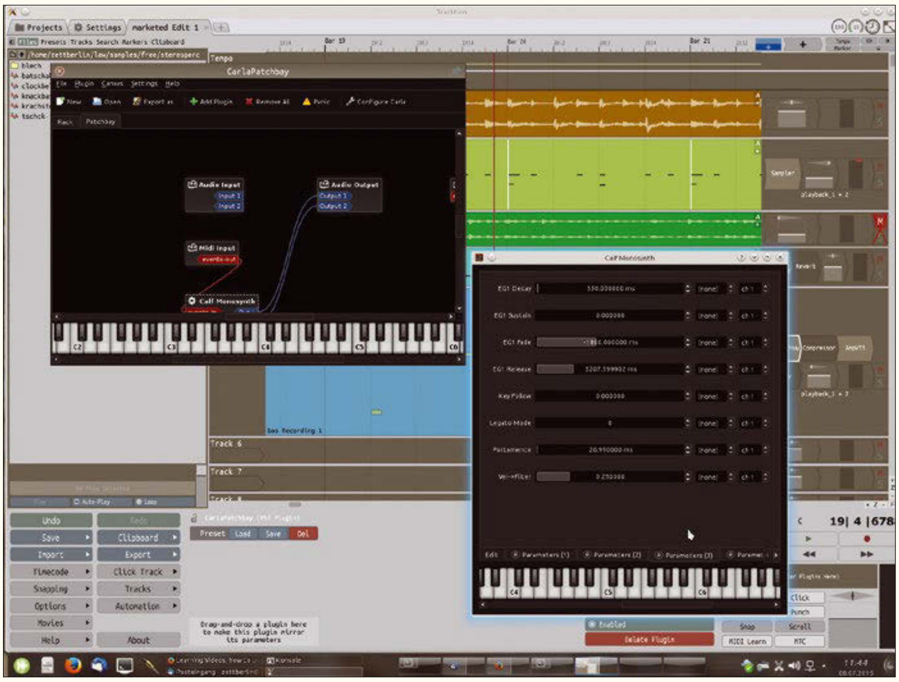


Figure 3: You can load plugins in the LV2 and Windows VST formats, thanks to Carla VST.

zoom factor into account: The greater the magnification, the finer the grid resolution. Although clips in a four-minute composition's overall view use full bars, they change to individual beats in a bar if you zoom in to just 10 seconds.

Notes in music actually follow strict mathematical rules regarding sequence and velocity. However, live music always deviates from this precision. Tracktion provides Groove Templates for its grid that emulate many shifts that are popular in practice. It is also possible to edit these templates in the Groove menus offered for MIDI and Step clips.

Most new functions for MIDI composers can be found in the Step sequencer clips (Figure 4). These are basically normal MIDI clips, but they specify the length of the notes and the loop and simplify the selection of the pitches.

A list in which an instrument is preset for each line appears on the left in a Step clip when you mouse over it. You can turn on an editor for velocity and duration of each note at the top, as needed. Using this approach, you can draw proper curves by holding the left mouse button or edit specific values for individual notes.

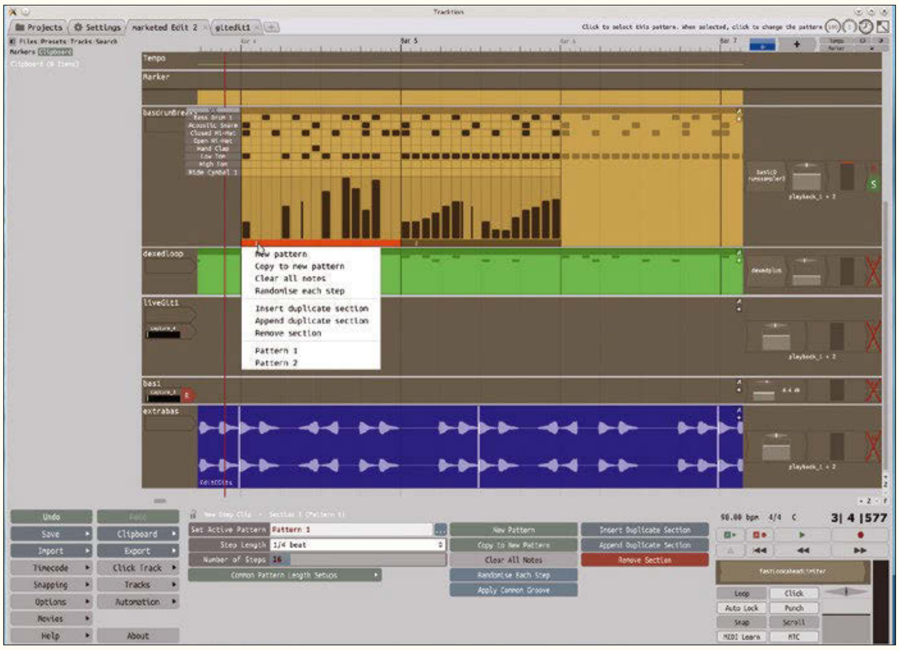
Clicking the name of the instrument opens a tool for setting a specific pitch at bottom center. Here you can choose one of the Groove Templates mentioned earlier, give it a new name, and draw common note sequences by clicking.

Besides setting the tone, the software also lets you select an individual tone generator in the new *Set Destination* menu. Here you can control different samplers and synthesizers simultaneously using a Step clip. This assumes that the different tone generators, which a Step clip will serve, are pooled in a rack plugin. The *Set Destination* menu displays a list of all the modules installed in a rack plugin that are able to receive MIDI notes.

Anyone wanting to play various instruments from a Step clip without using a rack plugin can use an old trick: Set the pitch on the keyboard below and one of the 10 MIDI channels for the instrument output in the Step clip. With instruments that also let you adjust these values, you can tailor them exactly to the instrument in the clip. Set the other sound generators so they don't output anything on that channel or the set pitch.

The developers have installed some wonderful innovations in the classic MIDI editor, which presents itself as a tool directly above the respective clip. Like Ardour, Tracktion T6 doesn't have a special editor in a separate window. If the appropriate zoom factor is set, this mode is good for direct editing and has the advantage that individual MIDI notes actually appear during composing in the position in the arrangement where the software plays them.

Tracktion recorded both the notes and the controller signals directly into the clip from the keyboard (Behringer UMX25) connected in the test. You can then edit them



Set Active Pattern: Pattern 1
Step Length: 1/4 beat
Number of Steps: 16

Figure 4: You can map the velocity as a kind of curve in Step sequencers. Selecting the pattern and duration of each note is pretty flexible here.

using a pen tool by clicking *Controller* or *Type* in the editor. You can draw the automation selected in the curve editor in Tracktion T6 for individual notes, as required (Figure 5). To do so, first select the note with a simple left-click, and then click *Show Note Automation* in the tool at the bottom in the middle.

For all its interesting features for loop-oriented, electronic music, the program is still meant primarily for classical audio recordings.

Tape Machine

The software is consistently nondestructive when editing: All cut and loop operations only affect the behavior of clips that are visible in the editor; the file with the original recording remains unchanged. Some operations are inherently not specifically applicable to these clips, especially those that need the original material to be recalculated.

For example, the *Loop Properties* mode doesn't work at all with the material from the clip itself, but rather with the respective source file. The same applies to the functions in *View Source Info* (Figure 6).

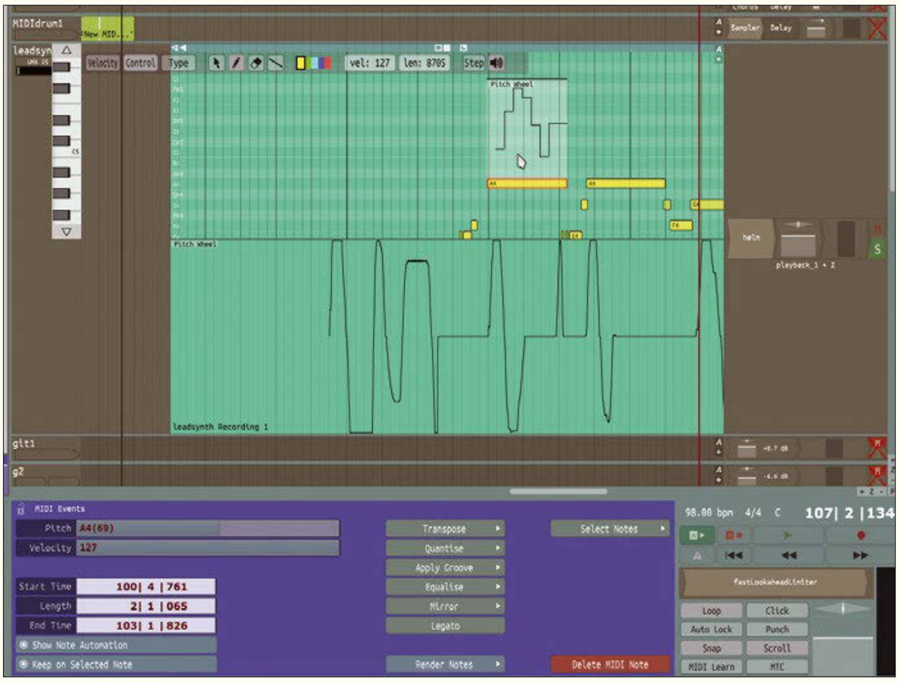
To target these special functions to just one clip, you need a new audio file that contains only the material in the clip. To create one, export the clip using the *Render Clip* tool at bottom right in the audio clip tool. The program collects these files below `Render` in the project folder. The original recording remains completely unchanged.

Tracktion isn't familiar with special Punch markers, which automatically insert a recording from a specific point in a piece of music. Instead, Punch is controlled by the same I/O auxiliary lines that also mark the beginning and end of loops. When playing, you can set the marks on the cursor's position with the *I* and *O* keys.

The mode for retrospective recording is an exciting new function for projects with live musicians. You can set up a time frame that the software uses to record all input in the menu bottom left in *Options* | *Retrospective Record*. Those who had a great idea while warming up on their guitar or MIDI keyboard no longer need to fret that the recorder wasn't running yet.

Clicking the button with the clock at the top right adds the material at the cursor position to the currently active track. The program takes the last recording input for a specific track into account. If you set a five-minute buffer and record first MIDI notes and then audio data in this period, you will be left with the audio data only. The function discards MIDI clips as soon as material appears on the audio track, and vice versa.

The way recordings are played can be manipulated automatically. To do this, simply drag the *A* icon at the top right end of the track onto an adjuster that you want



vel: 127
len: 8705

MIDI Events
Pitch A4(69)
Velocity 127
Start Time 100| 4 |761
Length 2| 1 |065
End Time 103| 1 |826

98.00 bpm 4/4 C 107| 2 |134

Figure 5: You can draw curves for automating individual notes in the Tracktion 6 MIDI editor.

to automate. Tracktion immediately draws the curve with the parameter data on the track. This feature is nice and compact and similar to the actual material, and thus to the music being played. Things can become a little confusing with multiple curves, however, which is why the program now provides its own automation tracks (Figure 7).

Those who like recording multiple takes spontaneously for a passage will appreciate the new comp groups. If you record the same take successively on different tracks in the same comp group, the software allows you to set sections in each track, which the program then plays exclusively. This mode lets you put together a quick and intuitive take from the best sections of various attempts.

Conclusions

Tracktion T6 is the first version for Linux that looks new and interesting at first glance. Its lean and fast base library is even recommended for computers and screens that are not powerful enough for fireballs like Bitwig Studio or Ardour.

You can buy a full-fledged studio that is outstandingly well integrated with Linux for just $60, and the program does not bombard users with advertising or use elaborate copy protection mechanisms, such as USB dongles. ■■■

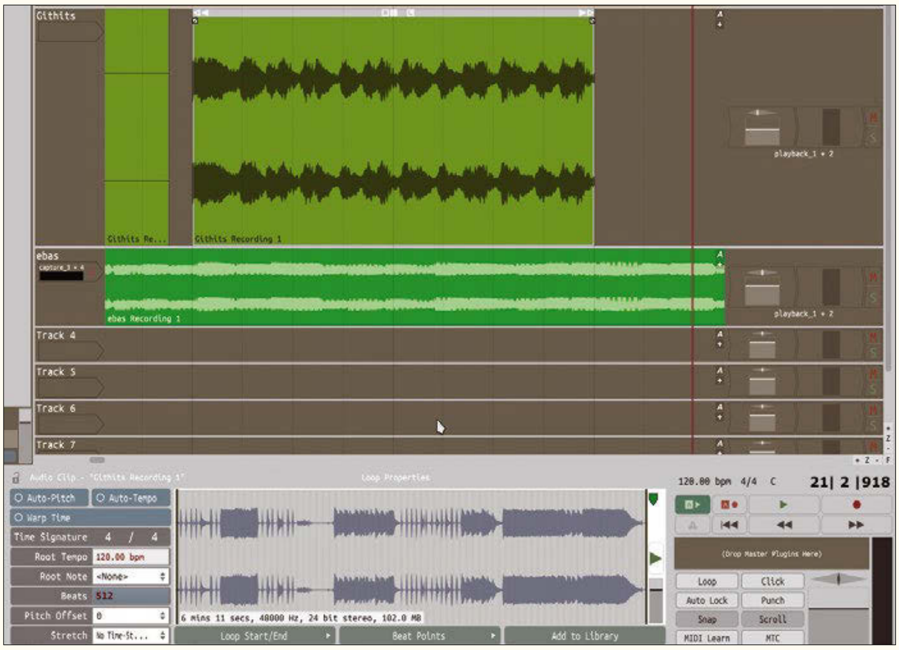

Time Signature 4 / 4
Root Tempo 120.00 bpm
Root Note <None>
Beats 512
Pitch Offset 0
Stretch No Time-St...
6 mins 11 secs, 48000 Hz, 24 bit stereo, 102.0 MB
120.00 bpm 4/4 C 21| 2 |918

Figure 6: The complete, six-minute-long source file is displayed on the *Loop Properties* tab for a clip that is only 20 seconds long.

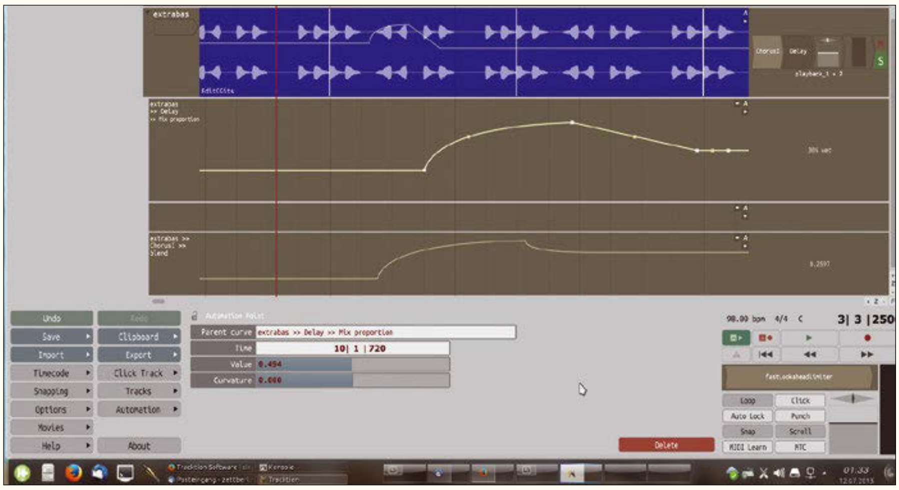

Automation Point
Parent curve extrabas >> Delay >> Mix proportion
Time 10| 1 | 720
Value 0.494
Curvature 0.000
98.00 bpm 4/4 C 3| 3 |250

Figure 7: Having the automation curves in their tracks helps provide a clearer overview in the software compared with several overlapping curves directly in the track.

INFO

[1] Tracktion T6: *http://www.tracktion.com*

[2] Juce Library: *http://www.juce.com*

[3] Download Juce: *http://www.juce.com/download*

[4] élastique Pro: *http://www.zplane.de/index.php?page=description-elastique*

[5] Marketplace: *https://marketplace.tracktion.com/app/*

Using Atom packages

Package Power

The Atom text editor's default functionality can be extended using packages. We look at packages that coders and writers alike may find rather useful. *By Dmitri Popov*

DMITRI POPOV

Dmitri Popov has been writing exclusively about Linux and open source software for many years, and his articles have appeared in Danish, British, US, German, Spanish, and Russian magazines and websites. Dmitri is an amateur photographer, and he writes about open source photography tools on his Scribbles and Snaps blog at *scribblesandsnaps.wordpress.com*.

Atom is a powerful and flexible text editor as it is, but thanks to its extensible architecture, you can teach it some useful tricks by installing additional packages. The official package repository [1] contains hundreds of modules. Some of them add very specific features, whereas others bring improvements that enhance the overall user experience and make coding and writing in Atom more efficient. Need packages like these? Then read on.

There's a Package for That

Sometimes a seemingly minor improvement can have a significant effect. The Seti Icons package [2] is a case in point. Once installed, this package replaces the default Atom icon set. At first sight, the change is purely cosmetic: The icon set uses a different color palette. This is an improvement in itself, but look closer, and you'll notice that this icon set does a much better job of differentiating files by their type (Figure 1). Although the default Atom icon set has one icon for all text file types, the Seti Icons package has dedicated icons for each file type. In practice, this means that you can immediately identify HTML, Markdown, and text files by their icons. This may not sound like much, but if you try the Seti Icons package, you might find it difficult to go back to the default icon set.

The Drag-and-Drop Text package [3] solves another small but important problem for users who prefer to use the mouse when working with text. By default, Atom doesn't support moving text selections with the mouse; however, the Drag-and-Drop Text package fixes this deficiency. Although it supports several actions, their behavior is slightly different from what you might expect. To drag and copy a text fragment, you make a selection, press and hold the left mouse button on the selected text, wait until you see a red border around the selection, drag the mouse to the desired location in the text, and release the button. To move a text fragment, you make a selection and hold the left mouse button on it until the selection disappears, drag the mouse to another location in the text, and release the button. If you release the left button before you move the mouse to the desired location, the described steps perform copy-and-paste and cut-and-paste actions.

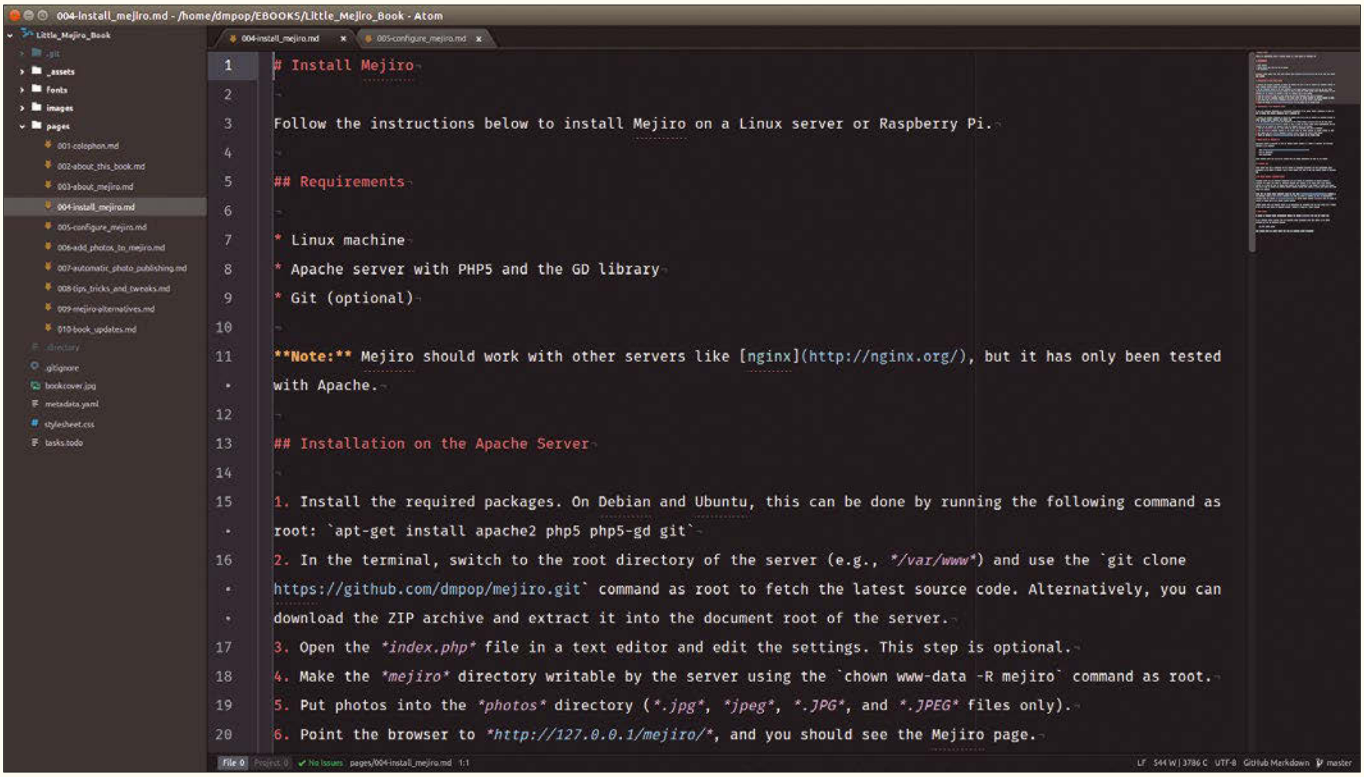
Atom supports a vast number of keyboard shortcuts (or keybindings), so you can perform practically any action without lifting your hands from the keyboard. The tricky part is to remember all the shortcuts, or at least the most useful ones. Keybinding Cheatsheet package [4] to the rescue. Once installed, it lists all actions and their keybindings in a separate sidebar (Figure 2). To toggle the Keybindings sidebar, use the Ctrl + Alt + / shortcut or run *Keybinding Cheatsheet: Toggle* in the Command Palette.

Versioning is an essential feature for many coders and writing professionals.

If you happen to use Git as your preferred version control system, then the Git History package [5] is right up your alley. For a less technical solution, the Local History package [6] is the way to go. As the name suggests, this package keeps a history of all edits for each file,

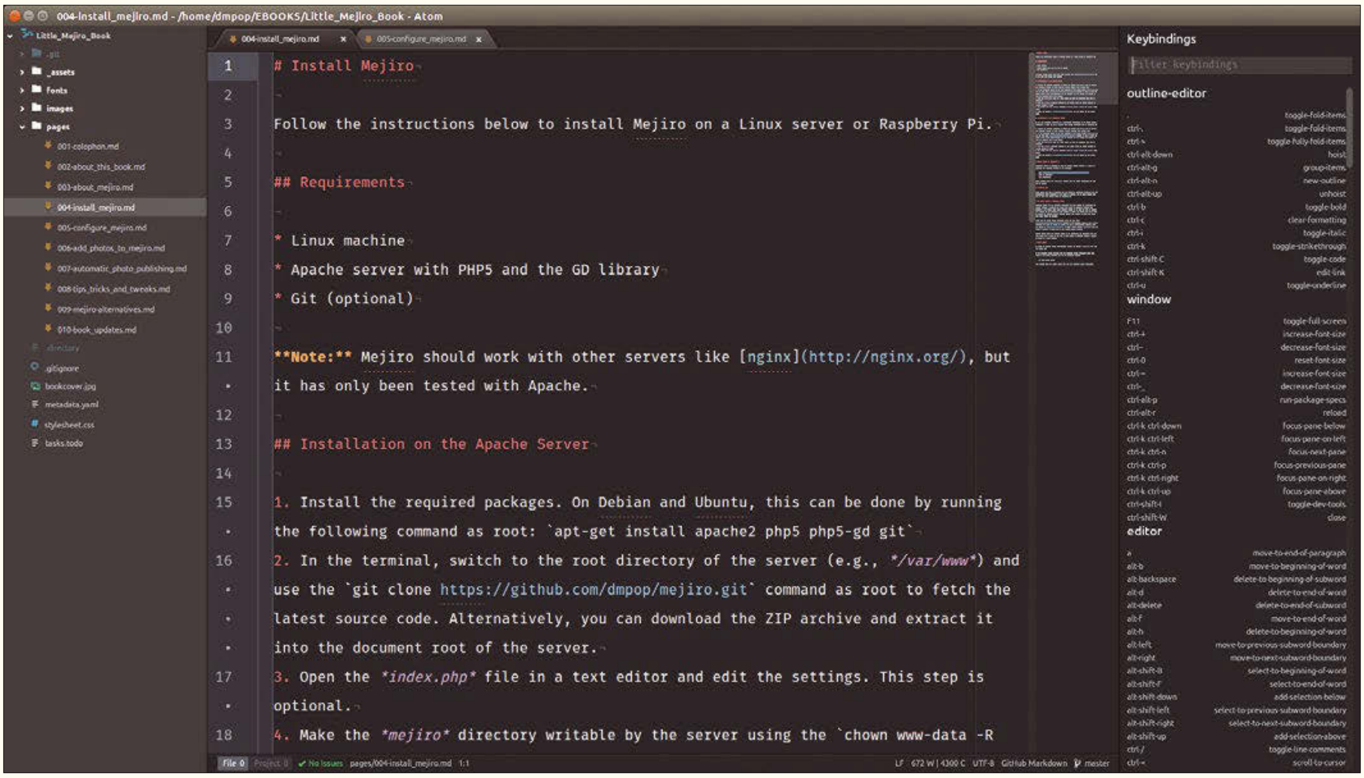
MANAGING PACKAGES IN ATOM

The *Install* section of the *Settings* interface makes it possible to find, install, and manage packages directly from within Atom. Use the search field to find packages matching a specific name, then hit the *Install* button next to the desired package to install it. To manage installed packages, switch to the *Packages* section. Here, you can configure, disable, and uninstall any package using the appropriate buttons next to it.



```
# Install Mejiro

Follow the instructions below to install Mejiro on a Linux server or Raspberry Pi.

## Requirements

* Linux machine
* Apache server with PHP5 and the GD library
* Git (optional)

**Note:** Mejiro should work with other servers like [nginx](http://nginx.org/), but it has only been tested
with Apache.

## Installation on the Apache Server

1. Install the required packages. On Debian and Ubuntu, this can be done by running the following command as
root: `apt-get install apache2 php5 php5-gd git`
2. In the terminal, switch to the root directory of the server (e.g., */var/www*) and use the `git clone
https://github.com/dmpop/mejiro.git` command as root to fetch the latest source code. Alternatively, you can
download the ZIP archive and extract it into the document root of the server.
3. Open the *index.php* file in a text editor and edit the settings. This step is optional.
4. Make the *mejiro* directory writable by the server using the `chown www-data -R mejiro` command as root.
5. Put photos into the *photos* directory (*.jpg*, *jpeg*, *.JPG*, and *.JPEG* files only).
6. Point the browser to *http://127.0.0.1/mejiro/*, and you should see the Mejiro page.
```

Figure 1: The Seti Icons package offers a better alternative to the default icon set.



```
# Install Mejiro

Follow the instructions below to install Mejiro on a Linux server or Raspberry Pi.

## Requirements

* Linux machine
* Apache server with PHP5 and the GD library
* Git (optional)

**Note:** Mejiro should work with other servers like [nginx](http://nginx.org/), but
it has only been tested with Apache.

## Installation on the Apache Server

1. Install the required packages. On Debian and Ubuntu, this can be done by running
the following command as root: `apt-get install apache2 php5 php5-gd git`
2. In the terminal, switch to the root directory of the server (e.g., */var/www*) and
use the `git clone https://github.com/dmpop/mejiro.git` command as root to fetch the
latest source code. Alternatively, you can download the ZIP archive and extract it
into the document root of the server.
3. Open the *index.php* file in a text editor and edit the settings. This step is
optional.
4. Make the *mejiro* directory writable by the server using the `chown www-data -R
```

outline-editor

.	toggle-fold-items
ctrl-.	toggle-fold-items
ctrl->	toggle-fully-fold-items
ctrl-alt-down	hoist
ctrl-alt-g	group-items
ctrl-alt-n	new-outline
ctrl-alt-up	unhoist
ctrl-b	toggle-bold
ctrl-c	clear-formatting
ctrl-i	toggle-italic
ctrl-k	toggle-strikethrough
ctrl-shift-C	toggle-code
ctrl-shift-K	edit-link
ctrl-u	toggle-underline

window

F11	toggle-full-screen
ctrl-+	increase-font-size
ctrl--	decrease-font-size
ctrl-0	reset-font-size
ctrl-=	increase-font-size
ctrl-_	decrease-font-size
ctrl-alt-p	run-package-specs
ctrl-alt-r	reload
ctrl-k ctrl-down	focus-pane-below
ctrl-k ctrl-left	focus-pane-on-left
ctrl-k ctrl-n	focus-next-pane
ctrl-k ctrl-p	focus-previous-pane
ctrl-k ctrl-right	focus-pane-on-right
ctrl-k ctrl-up	focus-pane-above
ctrl-shift-I	toggle-dev-tools
ctrl-shift-W	close

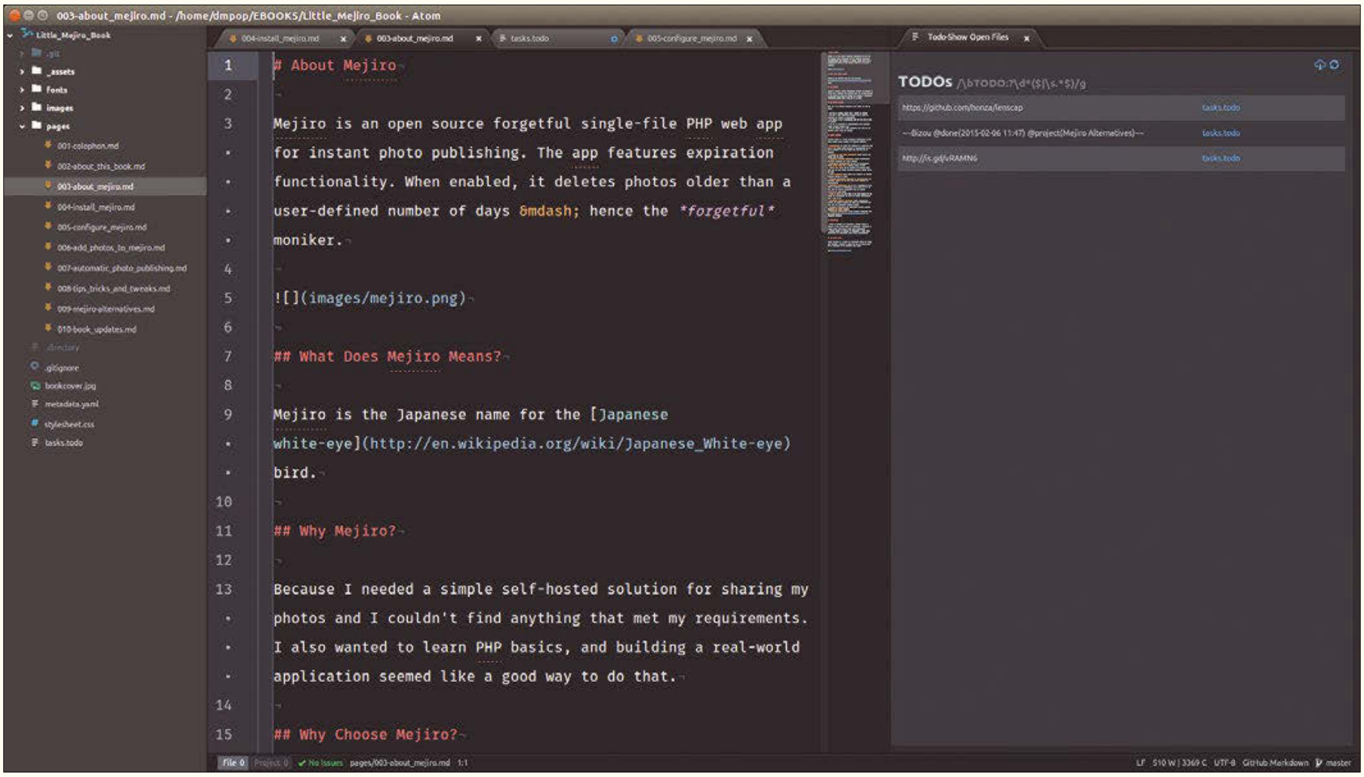
editor

a	move-to-end-of-paragraph
alt-b	move-to-beginning-of-word
alt-backspace	delete-to-beginning-of-subword
alt-d	delete-to-end-of-word
alt-delete	delete-to-end-of-subword
alt-f	move-to-end-of-word
alt-h	delete-to-beginning-of-word
alt-left	move-to-previous-subword-boundary
alt-right	move-to-next-subword-boundary
alt-shift-B	select-to-beginning-of-word
alt-shift-F	select-to-end-of-word
alt-shift-down	add-selection-below
alt-shift-left	select-to-previous-subword-boundary
alt-shift-right	select-to-next-subword-boundary
alt-shift-up	add-selection-above
ctrl-/	toggle-line-comments
ctrl-=	scroll-to-cursor

Figure 2: The Keybinding Cheatsheet package lists all of Atom's keybindings.

and you can easily view previous versions of the file as well as compare different versions using a so-called diff tool (by default, the package uses the Meld tool available in the software repositories of many Linux distributions). Local History stores file versions in the `~/.atom/local-history` directory, which makes it easy to create backups and extract the version you need manually.

Speaking of history, the Clipboard History package [7] can improve the default clipboard functionality by storing a list of all selections made during an editing session.



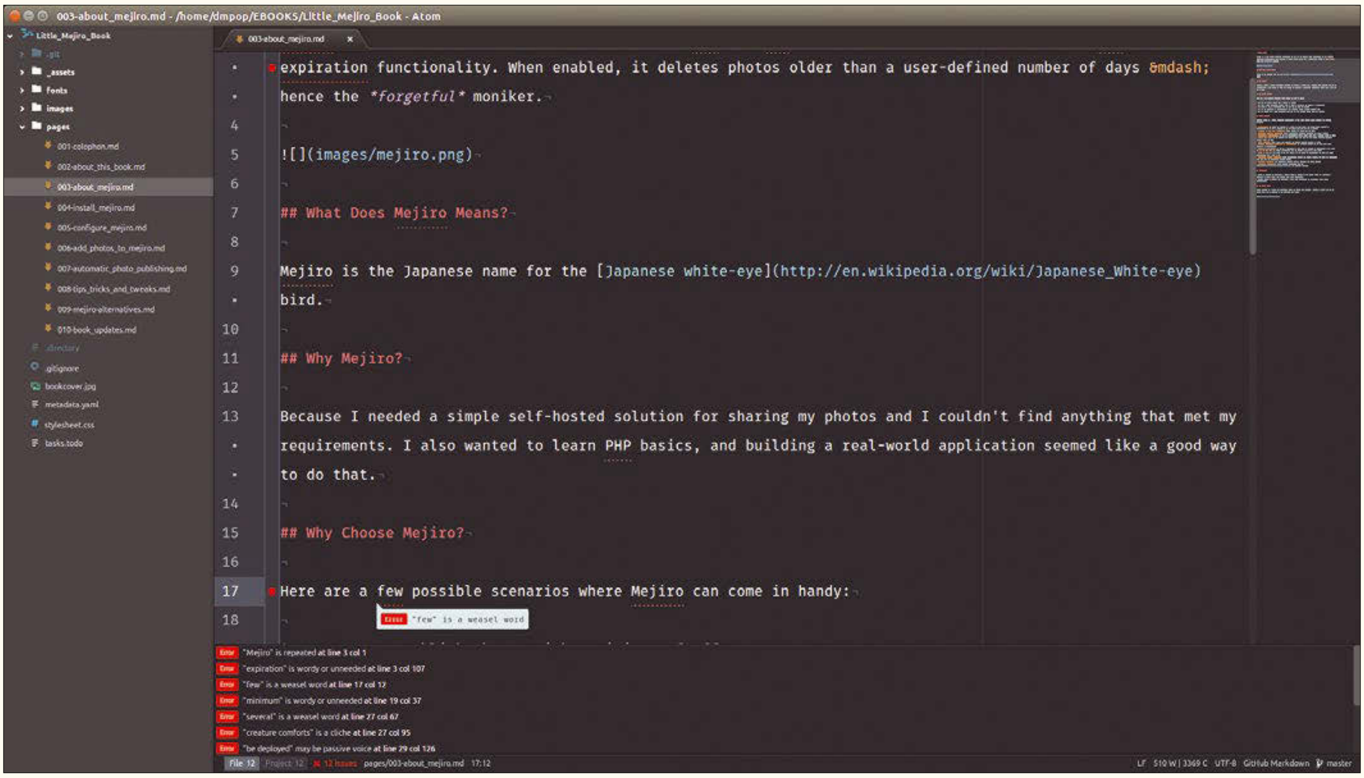
```
# About Mejiro

Mejiro is an open source forgetful single-file PHP web app
for instant photo publishing. The app features expiration
functionality. When enabled, it deletes photos older than a
user-defined number of days &mdash; hence the *forgetful*
moniker.

![](images/mejiro.png)

## What Does Mejiro Means?

Mejiro is the Japanese name for the [Japanese
white-eye](http://en.wikipedia.org/wiki/Japanese_White-eye)
bird.

## Why Mejiro?

Because I needed a simple self-hosted solution for sharing my
photos and I couldn't find anything that met my requirements.
I also wanted to learn PHP basics, and building a real-world
application seemed like a good way to do that.

## Why Choose Mejiro?
```

TODOs /\bTODO:?\d*($|\s.*$)/g

https://github.com/honza/lenscap	tasks.todo
---Bizou @done(2015-02-06 11:47) @project(Mejiro Alternatives)---	tasks.todo
http://is.gd/vRAMN6	tasks.todo

```
File 0  Project 0  ✓ No Issues  pages/003-about_mejiro.md  1:1        LF  510 W | 3369 C  UTF-8  GitHub Markdown  master
```

Figure 3: TODO Show can locate specific labels in files and projects.



```
expiration functionality. When enabled, it deletes photos older than a user-defined number of days &mdash;
hence the *forgetful* moniker.

![](images/mejiro.png)

## What Does Mejiro Means?

Mejiro is the Japanese name for the [Japanese white-eye](http://en.wikipedia.org/wiki/Japanese_White-eye)
bird.

## Why Mejiro?

Because I needed a simple self-hosted solution for sharing my photos and I couldn't find anything that met my
requirements. I also wanted to learn PHP basics, and building a real-world application seemed like a good way
to do that.

## Why Choose Mejiro?

Here are a few possible scenarios where Mejiro can come in handy:
                 Error "few" is a weasel word
```

```
Error "Mejiro" is repeated at line 3 col 1
Error "expiration" is wordy or unneeded at line 3 col 107
Error "few" is a weasel word at line 17 col 12
Error "minimum" is wordy or unneeded at line 19 col 37
Error "several" is a weasel word at line 27 col 67
Error "creature comforts" is a cliche at line 27 col 95
Error "be deployed" may be passive voice at line 29 col 126
File 12  Project 12  ✗ 12 Issues  pages/003-about_mejiro.md  17:12        LF  510 W | 3369 C  UTF-8  GitHub Markdown  master
```

Figure 4: Lint Write Good helps to improve writing by identifying common grammar and style issues.

The package requires no configuration (although it has a couple of settings you can tweak) and is dead easy to use: Press the Ctrl + Shift + V keyboard shortcut and select the desired selection item from the drop-down list.

It's common programming practice to use the `CHANGED`, `TODO`, and `FIXME` labels in comments to flag the parts of the code that have been modified, require additional work, and need fixing. However, nothing stops you from using this technique in regular text to mark paragraphs and text segments that require attention. Although you can use Atom's search functionality to find these labels in the text, The TODO Show package [8] provides a more elegant and efficient way of dealing with them.

This package uses a dedicated sidebar to list all marked fragments neatly grouped by the label. The package can find labels in the opened files as well as in all files in the current project. To toggle the sidebar, use the Shift + Alt + T keyboard shortcut or run the appropriate Todo Show command from the Command Palette.

The package uses regular expressions to find the default labels, but you can define custom regular expressions for other labels by modifying the basic rule that looks like this:

```
'/\\b@?TODO:?\\d*($|\\s.*$)/g'
```

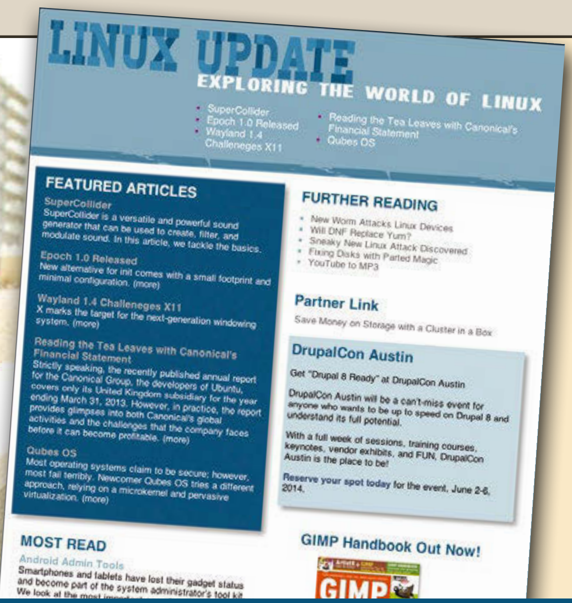
Adding the `COMMENT` label, for example, is a matter of replacing the `TODO` part of the rule as follows:

```
'/\\b@?COMMENT:?\\d*($|\\s.*$)/g'
```

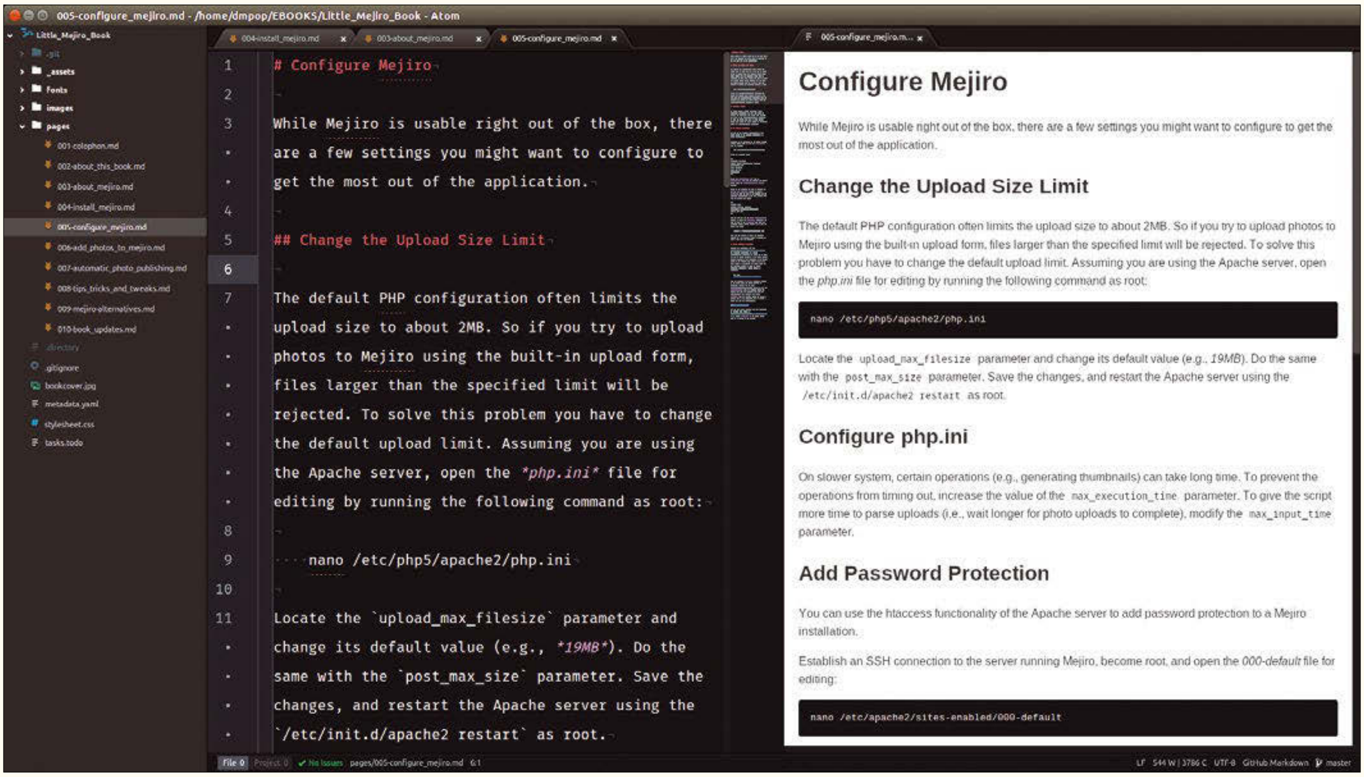
To add this rule to TODO Show, switch to the *Settings* | *Packages* section, press the *Settings* button next to the TODO Show item, and enter the rule into the *Find These Regexes* field (Figure 3).

GitHub Gist is a handy service for sharing code snippets and text fragments (gists in GitHub terminology), and the Gist It package [9] allows you to publish files and text selections directly from within Atom. By default, all gists are published anonymously, but you can change that by generating a GitHub token. To do this, point your browser to *github.com/settings/tokens/new* and generate a new token with the *gist* scope; then, paste the token into the *OAuth Token* field in the package's settings. To post the current file or text selection to Gist, use the Ctrl + Alt + G and Shift + Ctrl + Alt + G keyboard shortcuts, respectively.

Ft Photography, Fotolia



No matter how proficient you are at writing, the Linter Write Good package [10] can prove to be a useful addition to your toolbox. Although Linter Write Good won't magically transform your writing into a literary masterpiece, it can help you to identify and avoid common grammar and style issues (Figure 4). Among other things, the package can check for weasel words (e.g., somewhat, probably, some, most, etc.), use of passive voice, cliches, wordy phrases and unnecessary words, and adverbs that can weaken the meaning (really, very, extremely, etc.). To make Linter Write



```
# Configure Mejiro

While Mejiro is usable right out of the box, there
are a few settings you might want to configure to
get the most out of the application.

## Change the Upload Size Limit

The default PHP configuration often limits the
upload size to about 2MB. So if you try to upload
photos to Mejiro using the built-in upload form,
files larger than the specified limit will be
rejected. To solve this problem you have to change
the default upload limit. Assuming you are using
the Apache server, open the *php.ini* file for
editing by running the following command as root:

    nano /etc/php5/apache2/php.ini

Locate the `upload_max_filesize` parameter and
change its default value (e.g., *19MB*). Do the
same with the `post_max_size` parameter. Save the
changes, and restart the Apache server using the
`/etc/init.d/apache2 restart` as root.
```

Configure Mejiro

While Mejiro is usable right out of the box, there are a few settings you might want to configure to get the most out of the application.

Change the Upload Size Limit

The default PHP configuration often limits the upload size to about 2MB. So if you try to upload photos to Mejiro using the built-in upload form, files larger than the specified limit will be rejected. To solve this problem you have to change the default upload limit. Assuming you are using the Apache server, open the *php.ini* file for editing by running the following command as root:

```
nano /etc/php5/apache2/php.ini
```

Locate the `upload_max_filesize` parameter and change its default value (e.g., *19MB*). Do the same with the `post_max_size` parameter. Save the changes, and restart the Apache server using the `/etc/init.d/apache2 restart` as root.

Configure php.ini

On slower system, certain operations (e.g., generating thumbnails) can take long time. To prevent the operations from timing out, increase the value of the `max_execution_time` parameter. To give the script more time to parse uploads (i.e., wait longer for photo uploads to complete), modify the `max_input_time` parameter.

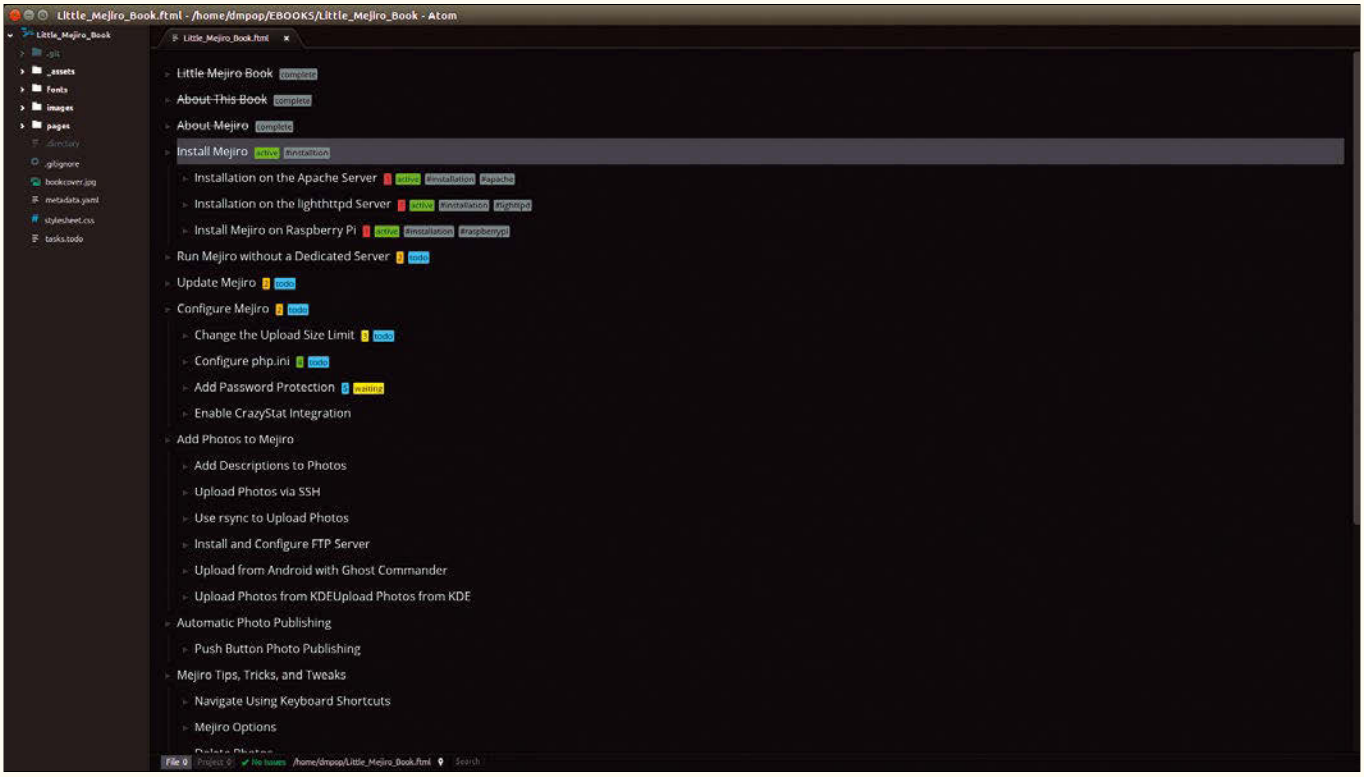
Add Password Protection

You can use the htaccess functionality of the Apache server to add password protection to a Mejiro installation.

Establish an SSH connection to the server running Mejiro, become root, and open the *000-default* file for editing:

```
nano /etc/apache2/sites-enabled/000-default
```

Figure 5: Markdown Scroll Sync keeps the original Markdown-formatted file and its preview in sync when scrolling.



- Little Mejiro Book complete
- ~~About This Book~~ complete
- About Mejiro complete
- Install Mejiro active #installation
  - Installation on the Apache Server 1 active #installation #apache
  - Installation on the lighthttpd Server 1 active #installation #lighttpd
  - Install Mejiro on Raspberry Pi 1 active #installation #raspberrypi
- Run Mejiro without a Dedicated Server 2 todo
- Update Mejiro 2 todo
- Configure Mejiro 2 todo
  - Change the Upload Size Limit 3 todo
  - Configure php.ini 4 todo
  - Add Password Protection 5 waiting
  - Enable CrazyStat Integration
- Add Photos to Mejiro
  - Add Descriptions to Photos
  - Upload Photos via SSH
  - Use rsync to Upload Photos
  - Install and Configure FTP Server
  - Upload from Android with Ghost Commander
  - Upload Photos from KDEUpload Photos from KDE
- Automatic Photo Publishing
  - Push Button Photo Publishing
- Mejiro Tips, Tricks, and Tweaks
  - Navigate Using Keyboard Shortcuts
  - Mejiro Options
  - Delete Photos

Figure 6: FoldingText transforms Atom into a capable outliner.

Good work, you also need to install the Linter package. Once enabled, Linter Write Good automatically parses the text, flags detected issues, and lists them in the bottom pane. You can then work through the list to fix the detected issues.

Markdown Creature Comforts

Markdown is a popular lightweight text formatting markup for authoring anything from README files and documentation to articles and ebooks. If it happens to be your preferred formatting option, you might find several handy Atom packages useful. Atom supports Markdown out of the box, and the editor allows you to display Markdown-formatted text and its preview side by side using the *Packages* | *Markdown Preview* | *Toggle Preview* command (or the Shift + Ctrl + M shortcut).

There is a tiny bump in the road, though: The editor doesn't keep the text and preview panes in sync, which can quickly become an annoyance when working with a long text requiring a lot of scrolling. The Markdown Scroll Sync package [11] fixes this issue (Figure 5). The Markdown Format [12] and Tidy Markdown [13] packages can help you to keep your Markdown-formatted text neat and clean by fixing common problems like double spaces, bad indentation, incorrect list numberings, and so on.

Do you need to generate a table of contents (TOC) for a long Markdown-formatted document? Markdown TOC [14] does the trick. This package supports several useful options, including TOC depth and automatic linking of TOC entries using anchor tags. Markdown TOC can also refresh the table of contents automatically every time you save the file.

Transform Atom into an Outliner

Designed to organize and manage text as a hierarchical tree, an outliner makes a perfect companion to a text editor. Instead of using a dedicated outliner, you can add outlining capabilities to Atom, courtesy of FoldingText [15]. Install the package, and you can create a new outline using the *File* | *New Outline* command (Figure 6). Mastering FoldingText's basics doesn't require a lot of effort. Press Enter to create a new entry in the current outline, hit Tab to indent the entry (i.e., move one level down), and use the Shift + Tab shortcut to un-indent the entry (i.e., move one level up). You can rearrange the entries in the outline by dragging them with the mouse, as well as collapse and expand the entries.

In addition to this basic functionality, FoldingText has a few clever tricks up its sleeve. The Ctrl + Space shortcut, for example, can be used to mark the current entry as completed, which means that you can use the outline as a simple way to manage tasks. FoldingText also makes it possible to assign tags, status labels, and priorities, and the package features the dedicated outline mode for speedy and efficient tagging and prioritizing.

To enter the outline mode, press the Esc key. You can then quickly assign tags to the currently selected entry by typing *t*. Each status in FoldingText has its own shortcut, too: Type *s t* for todo, *s w* for waiting, *s a* for active, and *s c* for complete. Assigning priorities in the outline mode is even easier: use number keys from *1* to *7* to add appropriate priority labels. FoldingText also features filtering functionality. Using it, you can limit the outline to a subset of entries matching a specific tag, status, or priority. To apply a filter, click on the desired tag, status, or priority label.

Package Syncing Made Easy

Finding and installing all the packages you need requires time and effort, so you wouldn't want to go through this process every time you install or reinstall Atom. Package Sync [16] provides the simplest solution to the problem. Install the package, and run the *Packages* | *Package Sync* | *Create Package List* command. This generates a list of all installed packages and saves it in the `~/.atom/packages.json` file.

Next time you need to install or reinstall the packages, place the generated `packages.json` file into the `~/.atom` directory, install Package Sync, and run *Packages* | *Package Sync* | *Sync*. ■■■

INFO

[1] Atom package repository: *atom.io/packages*

[2] Seti Icons: *atom.io/packages/seti-icons*

[3] Drag and Drop Text: *atom.io/packages/drag-drop-text*

[4] Keybinding Cheatsheet: *atom.io/packages/keybinding-cheatsheet*

[5] Git History: *atom.io/packages/git-history*

[6] Local History: *atom.io/packages/local-history*

[7] Clipboard History: *atom.io/packages/clipboard-history*

[8] TODO Show: *atom.io/packages/todo-show*

[9] Gist It: *atom.io/packages/gist-it*

[10] Linter Write Good: *https://atom.io/packages/linter-write-good*

[11] Markdown Scroll Sync: *atom.io/packages/markdown-scroll-sync*

[12] Markdown Format: *atom.io/packages/markdown-format*

[13] Tidy Markdown: *atom.io/packages/tidy-markdown*

[14] Markdown TOC: *atom.io/packages/markdown-toc*

[15] FoldingText: *atom.io/packages/foldingtext-for-atom*

[16] Package Sync: *atom.io/packages/package-sync5*

Using basic systemd commands

Control Plan

Systemd is a complex management structure with many commands and capabilities. We provide an overview of a few basic commands and their use. *By Bruce Byfield*

BRUCE BYFIELD

Bruce Byfield is a computer journalist and a freelance writer and editor specializing in free and open source software. In addition to his writing projects, he also teaches live and e-learning courses. In his spare time, Bruce writes about Northwest coast art. You can read more of his work at *http://brucebyfield.wordpress.com*

One of the criticisms of systemd, the new system management framework [1], is that it is a monolithic structure that violates the time-honored Unix principle of using small, dedicated tools. However, when you examine it, systemd proves less monolithic than it sounds. Instead, it has introduced a series of new tools for system administration, each of which has its own specific purpose.

What gives systemd a monolithic appearance is that its structure is consistent throughout. Users can be confident that a file ending in “ctl” is a command, whereas one ending in “conf” is a configuration file. Even more importantly, each command has the same structure:

```
COMMAND SUB-COMMAND OPTIONS
```

Often, options and sub-commands are the same for different commands.

Systemd tends to thoroughness, so it is impractical to list all of the main options and commands here. Instead, there is room only for some basic functions to give a sense of what each command can do. For more detailed information, consult each command’s man page or the systemd documentation [2]. You can get a sense of the complexity of systemd from the fact that `systemctl`, the most important systemd command, has a man page of 868 lines – more than four times as long as a command such as `cp` or `ls`. In this article, I’ll describe some of the basic commands introduced by systemd.

Systemctl

Systemctl is systemd’s main command. It manages units, which are the files to manage system resources. Units are organized into control groups, such as service, socket, device, mount, automount, and snapshot, which stores an image of the current state of systemd for later restoration. These control groups are used as an extension for unit files, so you always know which control group a unit belongs to. A system can have between 150 and 300 units, depending on its setup and what it runs. Usually, systems that run Gnome technologies have more units than KDE, possibly because systemd was initially developed with Gnome in mind.

The `systemctl` command manages units – enabling and disabling them, changing their state, and managing positive dependencies (what must be present for a unit to operate) and reverse dependencies (what cannot be present for a unit to operate). This broad array of controls makes `systemctl` extremely powerful – and, therefore, potentially capable of crashing your machine if used carelessly.

You can use `--all` (`-a`) to see a list of units as you compose the command structure (Figure 1). To avoid unintended consequences, whenever possible, add the option `--type=` (`-t`), to limit the command to particular control groups such as sockets or service, `--state=`, to specify whether units whose state is LOAD, SUB, or ACTIVE are affected, or both options. Use the command `list-dependencies` to view a unit's dependencies, and the `--reverse` option to list a unit's reverse dependencies.

Most of systemctl's basic functions are in its commands. With `show` and `status`, you can read current information about a unit or a comma-separated list of units. The `start` and `stop` commands activate a unit, whereas

```
set-property UNIT-NAME PARAMETER SETTING
```

edits a unit's definition.

Journalctl

Journals are one the more controversial features of systemd, because they are binary files rather than the traditional text files generally used in Linux. However, in compensation, the `journalctl` command has a thorough selection of choices for viewing logs.

Many distributions do not ship with systemd's journal enabled, so check to see if your installation has a `/var/log/journal` file. If not, you can set it up with the command:

```
setfacl -R -nm g:adm:rx,d:g:adm:rx /var/log/journal
```

You should also edit `/etc/systemd/journald.conf`, using the settings `storage=persistent` to save the journal permanently and `SystemMaxFileSize = 100M` so that the journal does not grow too large.

The bare command displays the journal, oldest entry first (Figure 2). However, for convenience, consider adding the option `--output=verbose` so that information is not cut off and `--reverse` (`-r`) so that the oldest entry is first.

You can filter journal messages with `--dmesg` (`-k`) to view only messages from the kernel, with `--system` for messages from system services, or with `--PID=` to view messages from a particular service. If you want to view messages from a particular book, run `journalctl` with `--system-boots`, then use the identifier in the second field of the output to run the records for the boot you want to examine. Alternatively, you can use `line=NUMBER` to view only a set number of events or use `--since` or `--until` to set a range of dates or times.

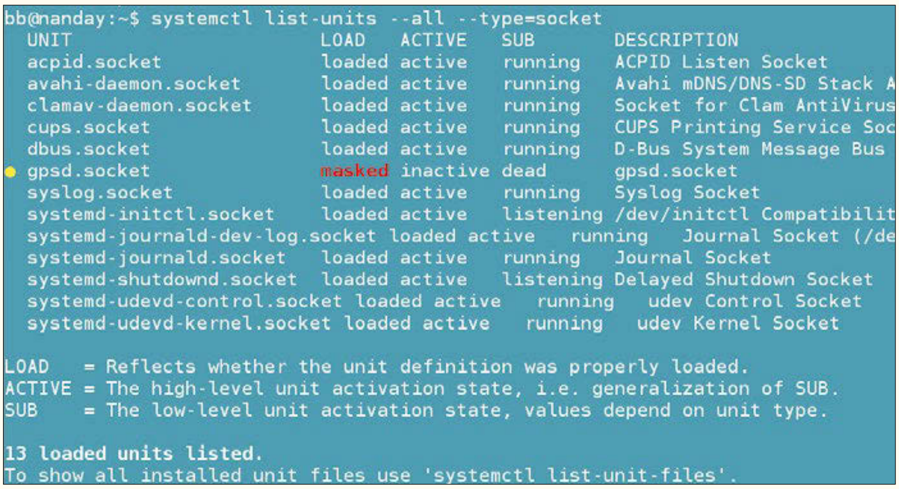
Halt, Poweroff, Reboot, Shutdown

These commands all close down the system. Technically, `halt` closes down the system without turning the power off, whereas `poweroff` and `shutdown` do turn the power off, and `reboot` does a cold reboot. However, all four commands include the options `--poweroff` and `--reboot`, so any distinction is blurred.

The `shutdown` command is unique in that options may be followed by a time to activate the command. The time is in hh:mm format, using a 24-hour clock. You also have the option of specifying a time in the future by specifying a number of minutes with a plus sign as a prefix. If no time is specified, then the default is `+1`. Using `+0` runs the command immediately. At the end of a shutdown command with a time argument, you can use the `wall` command to broadcast a message to all users.

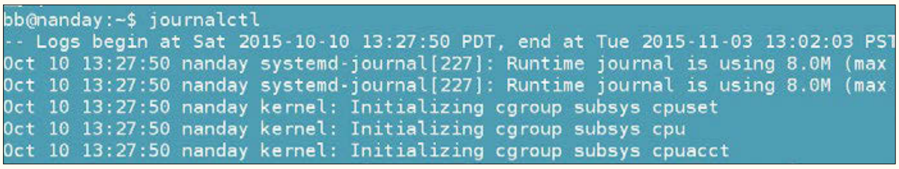
Hostnamectl

Systemd recognizes that hostnames have grown more varied over the years. The `hostnamectl` command recognizes three



```
bb@nanday:~$ systemctl list-units --all --type=socket
  UNIT                         LOAD   ACTIVE   SUB       DESCRIPTION
  acpid.socket                 loaded active   running   ACPID Listen Socket
  avahi-daemon.socket          loaded active   running   Avahi mDNS/DNS-SD Stack A
  clamav-daemon.socket         loaded active   running   Socket for Clam AntiVirus
  cups.socket                  loaded active   running   CUPS Printing Service Soc
  dbus.socket                  loaded active   running   D-Bus System Message Bus
● gpsd.socket                  masked inactive dead      gpsd.socket
  syslog.socket                loaded active   running   Syslog Socket
  systemd-initctl.socket       loaded active   listening /dev/initctl Compatibilit
  systemd-journald-dev-log.socket loaded active   running   Journal Socket (/de
  systemd-journald.socket      loaded active   running   Journal Socket
  systemd-shutdownd.socket     loaded active   listening Delayed Shutdown Socket
  systemd-udevd-control.socket loaded active   running   udev Control Socket
  systemd-udevd-kernel.socket loaded active   running   udev Kernel Socket

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

13 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'.
```

Figure 1: **Systemctl manages system resources, called units. Here, all socket units are listed.**



```
bb@nanday:~$ journalctl
-- Logs begin at Sat 2015-10-10 13:27:50 PDT, end at Tue 2015-11-03 13:02:03 PST
Oct 10 13:27:50 nanday systemd-journal[227]: Runtime journal is using 8.0M (max
Oct 10 13:27:50 nanday systemd-journal[227]: Runtime journal is using 8.0M (max
Oct 10 13:27:50 nanday kernel: Initializing cgroup subsys cpuset
Oct 10 13:27:50 nanday kernel: Initializing cgroup subsys cpu
Oct 10 13:27:50 nanday kernel: Initializing cgroup subsys cpuacct
```

Figure 2: **Systemd's journal begins by default with the oldest entry at the top of the list.**

types of hostname with its options: `--static` is the traditional hostname, used to initialize the system at bootup; `--transient` is the hostname assigned by a network; and `--pretty` is the high-level name for humans to read, which may contain many characters that static and transient hostnames are not permitted to use. Then, just to complicate matters further, `hostnamectl` can also adjust the hostname used in graphical interfaces. For example, `set-icon-name NAME` sets the hostname displayed on a desktop, and `set-chassis TYPE` designates a hostname as desktop, laptop, server, tablet, handset, or watch, to affect how they are displayed by some applications.

Use `set-hostname NAME` to change a hostname, modifying it with an option as necessary. To view the current hostname, use the command `status`. Add the option `--host` (`-H`) to change or view a hostname remotely. Note that a successful name change is marked only by a return to the command prompt.

Localectl

Locales are the languages and settings used for conventions such as currency and time and date formats. Locales typically begin with a two-letter lowercase abbreviation for the language, followed by an underscore and a two-letter uppercase abbreviation for the variant, and ending with an extension that indicates the character encoding – generally, `.utf8`. For example, `en_UK.utf8` stands for the English language as it is used in the United Kingdom. Keyboard maps specify the locale and the keyboard layout, such as Dvorak.

Under systemd, locales are managed by `localectl`. The `localectl` command can set the general system locale (`set-locale LOCALE`), the locale for the X Window system, which includes the general system locale (`set-x11-keymaps-variants`), and the keyboard locale (`set-keymap MAP TOGGLE`). If you need to look up locales, you can use the command `list-locales` for the display or `list-keymaps` for the keyboard.

Loginctl

To begin using `loginctl`, run the bare command to receive a list of current logins (Figure 3). If you want details for a session, add the sub-command `session-status`. Adding the session to a subcommand, you can activate a session, forcing it to replace the one currently displayed – for example, `lock-session`, `unlock-session`, `terminate-session`, or `kill-session`.

Besides controlling logins, `loginctl` can also be used to read what devices are being used by each account with `show-user USER` and `show-seat SEAT`.

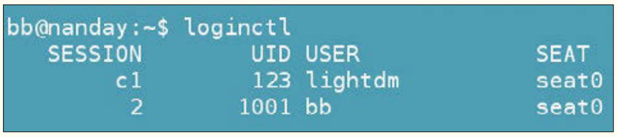
Systemd vs. Tradition

Learning systemd's commands reminds me of working with Debian's `dpkg-reconfigure` [3]. Both provide a consistent framework for configuring a Linux system. Of course, `dpkg-reconfigure` centers on packages, whereas systemd has more of a systems administrative perspective. With both, however, the consistency simplifies configuration and makes gathering information simpler.

In fact, exploring systemd's commands has at least partly reversed my reaction to it. When I first heard about systemd, it sounded like a massive complication that added an unneeded administrative layer to Linux. I suspected that, like the boot manager GRUB2 [4], it would scare novices away from hands-on administration – something that has always been one of the appeals of Linux.

However, now that I have actually looked into systemd, I think its consistency could actually encourage do-it-yourself discovery. Its common structure makes commands easier to learn, and users have a greater chance of guessing correctly if they cannot remember an option or command.

I am glad that most distributions have used aliases and symlinks to integrate systemd with existing tools for those who prefer not to learn it or want something to fall back on while they are learning. However, at least in some ways, systemd could actually be an improvement over the traditional tools. ■■■



```
bb@nanday:~$ loginctl
   SESSION        UID USER             SEAT
        c1        123 lightdm          seat0
         2       1001 bb               seat0
```

Figure 3: Loginctl controls the accounts logged in to the system.

INFO

[1] Systemd: *http://freedesktop.org/wiki/Software/systemd/*

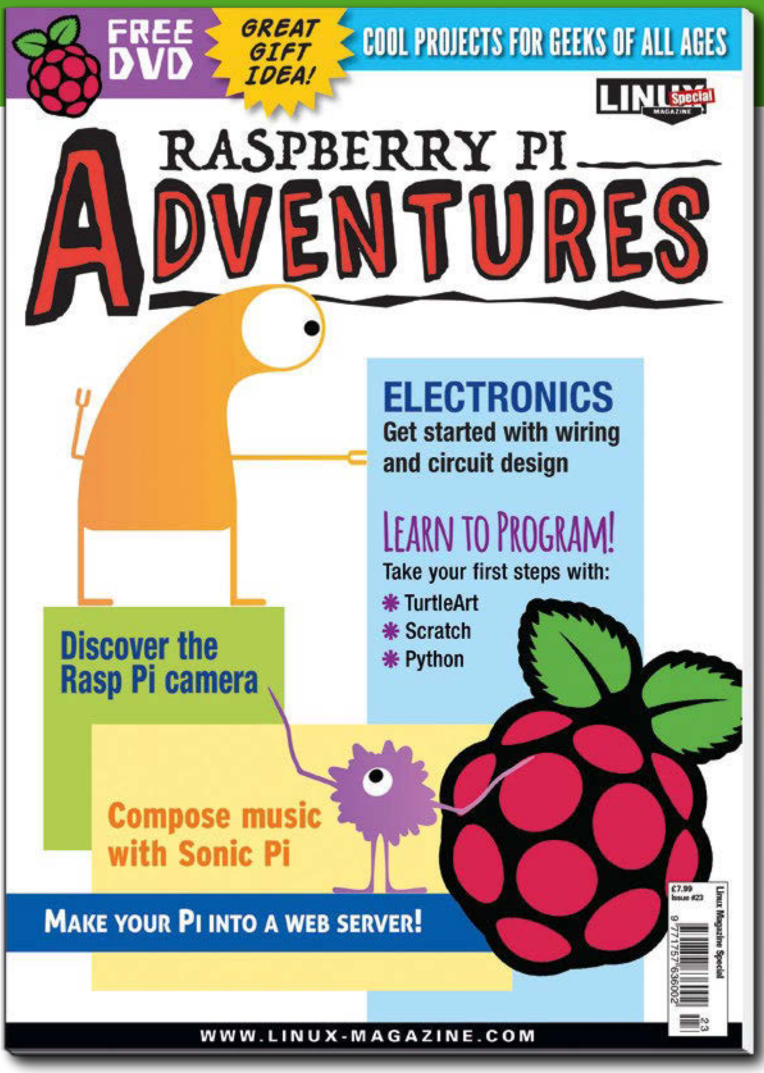
[2] Systemd man pages: *http://www.freedesktop.org/software/systemd/man/*

[3] Dpkg-reconfigure: *http://manpages.debian.org/cgi-bin/man.cgi?query=dpkg-reconfigure*

[4] GRUB2: *https://help.ubuntu.com/community/Grub2*

Shop the Shop shop.linuxnewmedia.com

RASPBERRY PI ADVENTURES


FREE DVD

GREAT GIFT IDEA!

COOL PROJECTS FOR GEEKS OF ALL AGES

LINUX MAGAZINE Special

RASPBERRY PI ADVENTURES

ELECTRONICS
Get started with wiring and circuit design

LEARN TO PROGRAM!
Take your first steps with:
- TurtleArt
- Scratch
- Python

Discover the Rasp Pi camera

Compose music with Sonic Pi

MAKE YOUR PI INTO A WEB SERVER!

£7.99 Issue #23

Linux Magazine Special

WWW.LINUX-MAGAZINE.COM

COOL PROJECTS FOR GEEKS OF ALL AGES

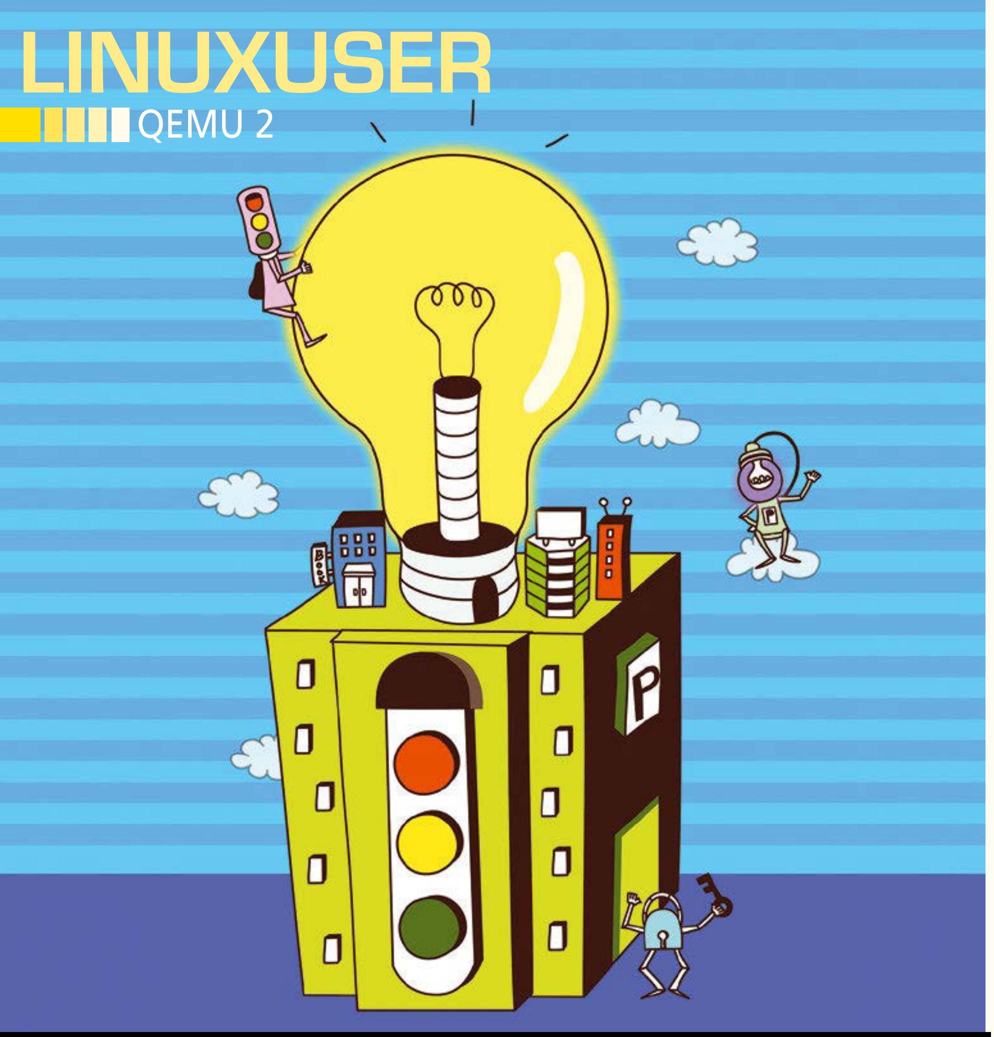
RASPBERRY PI ADVENTURES

is a one-volume special edition magazine for curious Raspberry Pi beginners. This easy, hands-on guide starts with an introduction to computers and offers a series of special hands-on projects illustrating many of the most popular uses for the Raspberry Pi.

ORDER YOUR VERY OWN ISSUE!

ORDER ONLINE:

shop.linuxnewmedia.com/se23



QEMU 2 as a versatile virtualization platform

Open Emu

The new version of QEMU is a free virtualization solution that offers excellent stability and flexibility. We show how to deploy QEMU 2 in a Live environment. *By Holger Reibold*

Many virtualization solutions exist today, but only a few can look back on a long tradition and many years of development work. After more than 10 years of development, the bell finally rings for round two of QEMU, which offers many special features for improved use and handling. This article looks at how to deploy QEMU 2 in a Live environment.

QEMU (Quick EMUlator) is a free emulator and virtualization solution that allows users to run a variety of operating systems in a virtual environment, much like many other virtualization systems. QEMU 2, which was released in April, can lay claim to several special features that you will not find in its competitors.

QEMU supports many options for launching the guest system, including multiprocessor emulation on single-processor systems. The open source software can also emulate other processor architectures such as PowerPC or ARM.

Because QEMU communicates directly with the kernel, the guest system runs at virtually the same speed as the host system (Figure 1). This good performance is achieved in combination with KVM, which makes it possible to run the guest at almost native speed. You only need to make sure that the processor you use supports hardware virtualization. However, this approach only works if the host and the guest share the same computer architecture. Another special feature is that QEMU does not need guest extensions for the guest, in contrast to VirtualBox or VMware. Additionally, the simulator integrates up to four hard disks.

Versatility

QEMU can be viewed as a technology pioneer in various areas. The developers started to leverage the benefits of KVM back in version 0.12, which helps achieve considerable performance boosts on Linux systems.

Like its competitors – whether free or commercial – QEMU runs on all popular operating systems and processor architectures. If you use QEMU without an accelerator, you do not even need administrative privileges. You can thus store a virtual machine including QEMU on a medium and run it on another computer.

The free virtualization software supports snapshots; thus, it can create multiple copies of the states of your virtual machines and revert system changes if needed. QEMU has other special features to offer: For example, the software supports live migration, system debugging, and booting from older disk formats. You can even emulate hardware errors.

QEMU has a flexible tool in the form of `qemu-img` for creating, converting, and encrypting image files (i.e., virtual hard disks in various formats). QEMU lets you boot from image files from other virtualization tools, and it provides support in the opposite direction, too: You can create virtual machines for KVM, Xen, and other hypervisor systems. It is even possible to export image files across the network. To do so, use the `qemu-ndb` tool, which in turn relies on the Network Block Device (NBD) protocol.

QEMU also supports libvirt, which is a cross-hypervisor abstraction layer for managing virtual machines. Building on this layer, management tools such as Virtual Machine Manager or `virsh` can manage a variety of virtualization solutions. Thus, there's nothing to prevent you from running QEMU parallel to existing virtualization solutions.

Getting Started with QEMU

In principle, you can run QEMU 2 in Windows and UNIX-based environments, but to leverage its full potential, you will probably want to opt for a Linux system. Debian or Ubuntu are good choices.

Before you start installing QEMU 2, you must explicitly enable support for hardware virtualization in your computer's BIOS. The settings differ greatly from BIOS to BIOS and between motherboard manufacturers. If you have a gigabyte motherboard, you will find the options for enabling the CPU's virtualization feature in the Award BIOS below *Advanced CPU Features*. For other motherboard manufacturers, refer to your motherboard documentation. You can check the CPU to see whether your CPU supports the required virtualization technologies using the following command:

```
grep "vmx" /proc/cpuinfo
```

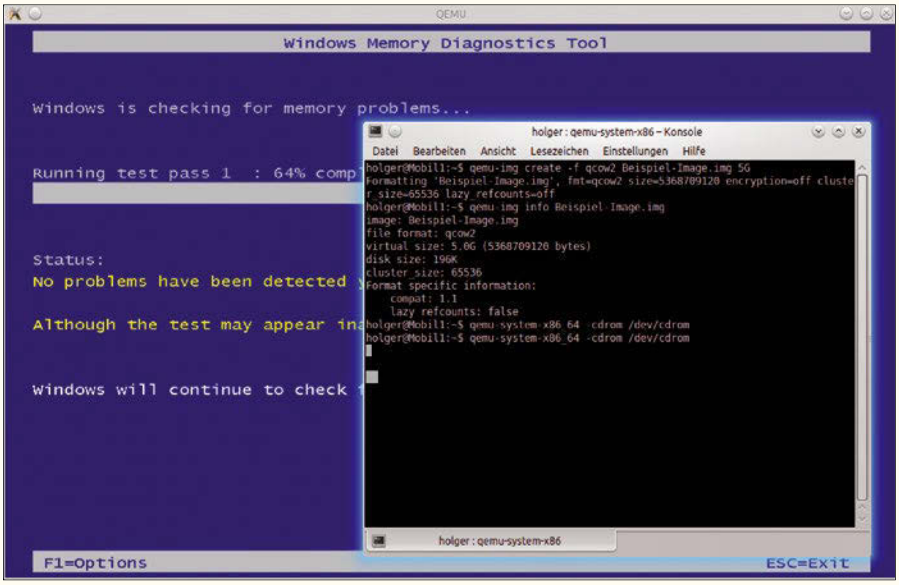
Once you have sorted out the question of virtualization support, you can proceed with the installation. More recent Linux distributions will group QEMU and KVM in the *qemu-kvm* package. However, because this package only supports x86 guest systems, it is a good idea to install the *qemu-kvm-extras* package, which provides emulators for other architectures.

The easiest way to install QEMU 2 is to use your distribution's package manager. Thus, if you work with Debian and Ubuntu, you would choose Synaptic. The important thing here is to enable all the program sources in the *System* | *Administration* | *Software sources* menu. If you prefer working at the command line, run the following command:

```
sudo apt-get install kvm qemu-kvm qemu-kvm-extras
```

On Fedora, Red Hat Enterprise Linux, and CentOS, you would use `yum` for the install:

```
yum install qemu-kvm qemu-kvm-extras
```


QEMU
Windows Memory Diagnostics Tool
holger : qemu-system-x86 – Konsole
Datei Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
holger : qemu-system-x86
F1=Options
ESC=Exit

```
Windows is checking for memory problems...

Running test pass 1  : 64% comp

Status:
No problems have been detected y

Although the test may appear ina

Windows will continue to check t
```

```
holger@Mobil1:~$ qemu-img create -f qcow2 Beispiel-Image.img 5G
Formatting 'Beispiel-Image.img', fmt=qcow2 size=5368709120 encryption=off cluste
r_size=65536 lazy_refcounts=off
holger@Mobil1:~$ qemu-img info Beispiel-Image.img
image: Beispiel-Image.img
file format: qcow2
virtual size: 5.0G (5368709120 bytes)
disk size: 196K
cluster_size: 65536
Format specific information:
    compat: 1.1
    lazy refcounts: false
holger@Mobil1:~$ qemu-system-x86_64 -cdrom /dev/cdrom
holger@Mobil1:~$ qemu-system-x86_64 -cdrom /dev/cdrom
```

Figure 1: Windows systems can be run with excellent performance.

Installation is also simple on openSUSE, where you turn to YaST and look for the *qemu-kvm* and *qemu-kvm-extras* packages.

If you are installing QEMU 2 with the help of the package manager, it makes sense to install a graphical interface for the virtualization environment at the same time (Figures 2 and 3). Doing so provides the convenience that may be familiar from using virtualization programs such as VirtualBox on Windows. This gives you a complete virtualization environment, and you can fire up the first virtual machine.

QEMU Fast Track

Once the basic QEMU 2 system is installed, you can install the first virtual machine, which typically involves three steps. In the first step, you generate a virtual hard disk – or an image file to be more precise. The second step is launching QEMU or KVM along with the virtual machine and executing the installation medium. This configures a virtual machine with the intended start options. The third step is installing the operating system. You only need to make sure that the emulated hardware is supported. There is no need to install special drivers on the guest system.

The virtualization environment uses identical start options on all operating systems. You can access and execute the most important options at the command line. The benefit of this is that any error messages will be output directly in the terminal.

To install a guest operating system, you first create the virtual hard disk. To do so, run the `qemu-img` tool that comes with the QEMU package using the `create` parameter. The `-f` option lets you determine the format of the image file:

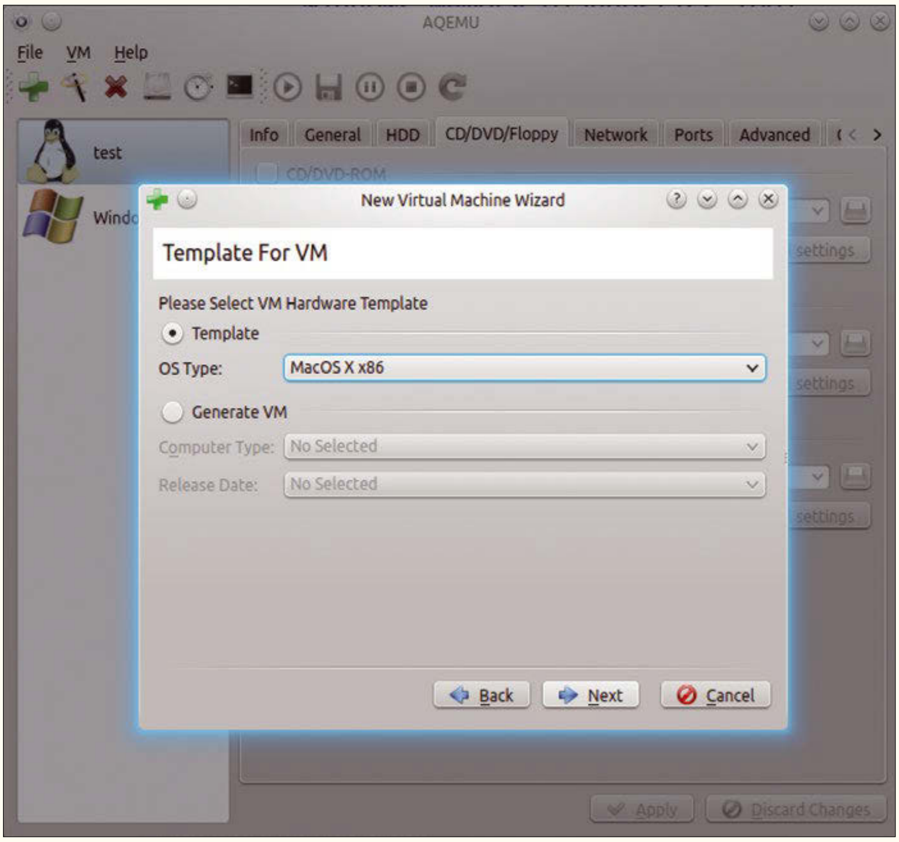
```
qemu-img create -f qcow2 example-image.img 5G
```

Next, you need a bootable image or medium for the operating system that you want to install as the guest system. You can call QEMU 2 with one of the following commands:

```
qemu-system-x86_64
qemu-system-i386
```

On some Linux distributions, you can also use the `kvm` command. If you are booting a virtual machine from a CD, remember to add the `-cdrom` option after the image file name. Pass in the name of the virtual hard disk you created with the `-hda` option. Now the virtual computer just needs to know that you want it to boot from the virtual CD, and the `-boot d` option clarifies this:

```
qemu-system-x86_64 -m 512 -hda example-image.img -cdrom example-OS.iso -boot d
```



Template For VM

Please Select VM Hardware Template

● Template

OS Type: MacOS X x86

○ Generate VM

Computer Type: No Selected

Release Date: No Selected

Figure 2: The AQEMU GUI has the greatest convenience factor.

Occasionally you might see an error message at boot time stating that the `bios.bin` file is missing. In this case, you need to add the `-L` start option along with the path to this file. If the `bios.bin` file resides in the current directory, you need to specify this with a dot:

```
qemu-system-i386 -m 512 -hda example-image.img -cdrom example-OS.iso -boot c -L .
```

This command launches the virtual PC in a separate window. When you click on the window, the mouse pointer is captured and you can use the mouse within the guest system. The Ctrl + Alt keyboard combination releases the mouse pointer again.

After completing the installation of the operating system, you need to restart the virtual machine; then, to boot the newly installed system from the virtual hard disk, use the `-boot c` option:

```
qemu-system-i386 -m 512 -hda example-image.img -cdrom example-OS.iso -boot c
```

You can also boot off the network, with:

```
qemu-system-i386 -m 512 example-OS.img -net user -net nic,model=pcnet
```

Here, QEMU 2 uses the integrated DHCP server to define a guest system's network settings.

QEMU Monitor – Controlling the Virtual Machine

To control your virtual machines at run time, use QEMU Monitor. The software is controlled by keyboard shortcuts and offers a versatile feature scope. You can use it to reject or replace removable media, freeze the state of a virtual machine, reactivate the machine as needed, and backup and restore states.

QEMU Monitor also lets you inspect the state of a virtual machine and migrate a virtual machine to another host. The tool even lets you modify the hardware and trigger emulated hardware errors.

After launching QEMU, press Ctrl + Alt + 2 to change to QEMU Monitor. (See Table 1 for a list of useful keyboard shortcuts.) The `info kvm` command tells you whether KVM hardware virtualization is enabled. The typical output looks like this:

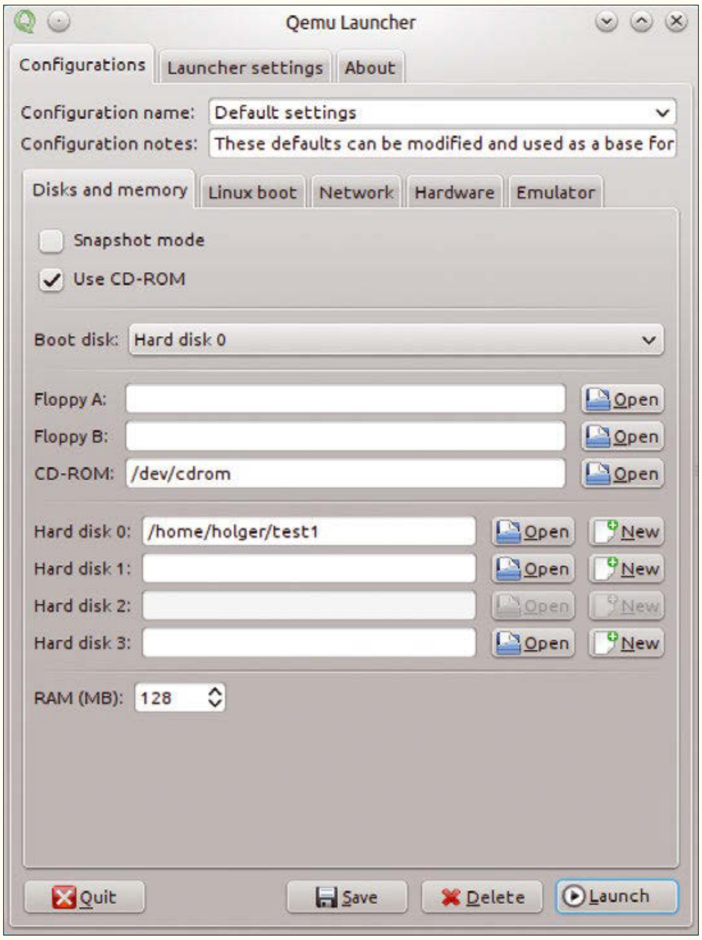
```
(qemu) info kvm
kvm support: enabled
```

Using the `history` parameter, you can output the command history:

```
(qemu) info history
0: 'help'
1: 'info'
2: 'info version'
3: 'info kvm'
4: 'info history'
```

To terminate an instance, without shutting down the guest system before doing so, simply use the `quit` command. This is equivalent to pressing the off button on the computer and therefore can cause loss of data:

```
(qemu) quit
```



Configuration name: Default settings
Configuration notes: These defaults can be modified and used as a base for

□ Snapshot mode
☑ Use CD-ROM

Boot disk: Hard disk 0

Floppy A:
Floppy B:
CD-ROM: /dev/cdrom

Hard disk 0: /home/holger/test1
Hard disk 1:
Hard disk 2:
Hard disk 3:

RAM (MB): 128

Figure 3: QEMU Launcher provides a basic QEMU GUI on most distributions.

If you want to reset a virtual machine, use the `system_reset` command as follows to do so:

```
(qemu) system_reset
```

The QEMU environment can also gracefully shut down the installed guest system. For this to happen, the guest needs to understand and interpret ACPI commands. The simulator uses the `system_powerdown` command to send an appropriate ACPI signal to the guest. In the case of a Linux guest system, you need to install the *acpid* package:

```
(qemu) system_powerdown
```

To pause an instance, use the `stop` command. You can look at the status of a (paused) instance by using the `info status` command, as shown in the following example:

```
(qemu) stop
(qemu) info status
VM status: paused
```

To run the instance again, use the `cont` option. Again, you can verify the status with `info status`:

```
(qemu) cont
(qemu) info status
VM status: running
```

QEMU has a very practical protection mechanism that fields various keyboard combinations instead of passing them through to the guest system. This is true, for example, of the keyboard combination Ctrl + Alt + Del. QEMU Monitor offers you an option for passing through these commands, though. For example, if you want to pass through the combination mentioned above, the following input will do the trick:

```
(qemu) sendkey ctrl-alt-delete
```

If you want to create screenshots of the guest, the `screendump` command will help. This command creates a PPM file:

```
(qemu) screendump screenshot.ppm
```

To eject a medium from the CD-ROM drive, use:

TABLE 1: Important Keyboard Shortcuts for QEMU Monitor

Keyboard Shortcut	Action
Ctrl + Alt	Releases the mouse and keyboard.
Ctrl + Alt + 1	Changes to the guest operating system's display.
Ctrl + Alt + 2	Changes to console 2, QEMU Monitor.
Ctrl + Alt + 3	Changes to console 3, serial output.
Ctrl + Alt + 4	Changes to console 4, parallel output.
Ctrl + Alt + H	Outputs the help with the option `-nographic`.
Ctrl + Alt + F	Toggles between full-screen and window mode.
Ctrl + Alt + +	Enlarges the screen output.
Ctrl + Alt + -	Reduces the size of the screen output.

```
(qemu) eject cdrom
```

If you need more options, you can type help to view the complete list of available commands for QEMU Monitor.

Accessing Storage Media

For a QEMU 2 instance to be able to access storage media, a medium first must be registered in the virtualization environment. To access media that already exist and query their status, you can use the info block command in QEMU Monitor. This lists the names and states of the storage media. If the input contains notes to the effect of *ide-hd* and *ide-cd* or *scsi-hd* and *scsi-cd*, this means that you are emulating IDE and SCSI hard disks and DVD/CD drives.

To access a virtual hard disk, you can use the options -hda <file>, -hdb <file>, -hdc <file>, -hdd <file>, and -drive, where <file> typically refers to an image. For example:

```
qemu-system-x86_64 -hda Hard_disc1.img -hdb Hard_disc2.img
```

If you are only using one virtual hard disk, you can dispense with the -hd*n* option:

```
qemu-system-x86_64 Hard_disc1.img
```

To access a (virtual) CD/DVD drive, use the options -cdrom <file> and -drive, where <file> refers to an image file or a physical device:

```
qemu-system-x86_64 -hda disc.img -cdrom cd.iso
```

If the host system has a CD drive, you can pass it through to the guest. If the host is a Linux system, the drive will be, for example, /dev/cdrom or /dev/dvd. To boot from the CD drive in this scenario, use the following command:

```
qemu-system-x86_64 -cdrom /dev/dvd
```

If you are using a Windows system as the host, you need the drive letter to integrate the physical drive. For example, the following command boots a CD/DVD from drive C:

```
qemu-system-x86_64 -L . -cdrom c:
```

You can also protect the storage media you use against changes. To do so, use the -snapshot option, which ensures the changes are not written to the storage medium itself but to temporary files. However, note that any changes you make are lost when you terminate the virtual machine:

```
qemu-system-x86_64 disc.img -snapshot
```

The info block commands tells QEMU Monitor to show you the temporary files:

```
(qemu) info block
ide0-hd0: removable=0 io-status=ok file=/tmp/a2.2B4u3P backing_file=disc.img ↩
  ro=0 drv=qcow2 encrypted=0
```

To store changes on the storage medium, use Ctrl + Alt + S. You can also use the commit command to store the changes in QEMU Monitor. The additional all parameter stores the changes on all connected drives:

```
(qemu) commit all
```

Under normal circumstances, you will typically want to write content to one drive in a targeted way. To save the changes on your first hard disk, run:

```
(qemu) commit hda
```

Because running the `commit` command is typically time consuming – the guest system is frozen to do this – you can use the `no-shutdown` option instead of the `-snapshot` option.

QEMU and KVM are capable not only of creating new virtual storage media but also of converting storage media for other virtualization programs, such as VMware. The `qemu-img` tool is used for this purpose; it supports all relevant image formats such as RAW (the default format), VMDK (VMware), VDI (VirtualBox), DMG (Mac image file), HDD (Parallels), and many more.

Storing Your Own Images

You can use QEMU for more than just executing virtual systems; it can also create an image – and in all of the popular formats. For example, you can quickly and easily create an image file and then provide it to third parties.

To do this, QEMU again uses the integrated command-line tool `qemu-img`. The `create` parameter lets you create your own image files. You need to specify the file name of the image file and the virtual size, as in the following example:

```
qemu-img create own_image.img 1G
Formating 'own_Image.img', fmt=raw, size=1048576 kB
```

After creating an initial image, you can use the `info` parameter to access the data for the image file:

```
qemu-img info own_image.img
image: disc.img
file format: raw
virtual size: 1.0G (1073741824 bytes)
disk size: 0
```

If you just want to exchange the image between QEMU installations, your best bet is to use the `qed` and `qcow2` formats. In particular, `qed` is optimized for fast access.

Besides letting you create images, `qemu-img` can also convert, compress, encrypt, and resize these images. To convert an image file to a different format, you would use `convert`; the `-O` parameter defines the target format. The conversion mechanism typically identifies the original format, but you can also state it explicitly with the `-f` parameter. If you want to convert a virtual hard disk in `raw` to the `qcow2` format, use the following command:

```
qemu-img convert -f raw -O qcow2 source-image.img target-Image.img
```

The `qemu-img` command also lets you encrypt an image. However, this only works if you use the image formats `qcow` and `qcow2`, as in the following:

```
Host ~$ qemu-img convert -O qcow2 -o encryption original-image.img encrypted_image.img
Disk image 'encrypted_image.img' is encrypted.
password: ********
```

QEMU uses a 128-bit AES encryption key.

QEMU for Advance Users

As with comparable virtualization solutions, the guest and host can establish reciprocal network connections. But, of course, this also works between guests, assuming you are running multiple instances. All you need to do to accomplish this is to configure the virtual network card with the `-net nic` option:

```
qemu-system-x86_64 first_instance.img -net nic
```

The `info network` command lets you retrieve information about the instance's VLANs and the matching devices, as in the following example:

```
(qemu) info network
VLAN 0 devices: user.0: net=10.0.2.0, restricted=ne1000.0: model=e1000,↩
  mac-addr=44:55:00:12:34:56
```

QEMU has an integrated DHCP server, which supports automatic network configuration of the guest systems on the user mode network stack. For example, all network-capable guest operating systems can establish an external connection immediately after provisioning. If you enable the `-net nic -net user` options, this gives you an internal DHCP server (10.0.2.2) and a DNS server (10.0.2.3) on the internal network (10.0.2.0).

Another special feature of QEMU 2 is its support for live migration, which involves transferring running virtual machines from one host system to another. The QEMU developers recommend using an identical hardware configuration on the source and target systems.

A live migration comprises three steps. First, you need to execute the source instance on the source host. Second, you start a new instance on the target host using the same parameters. The important thing to remember here is to use the `-incoming tcp:ip:port` option. Third, you need to run the `migrate -d tcp:ip:port` command in QEMU Monitor on the source host.

Conclusions

QEMU 2 gives users an excellent virtualization environment that does justice to even the strictest requirements for stability and flexibility. The program impresses in particular in performance, with another bonus in the comprehensive documentation on the project website. ■■■

What happens when something breaks, and there's no one left to fix it?

Auld Lang Syne

The passing of the first generation of programmers brings to light the predicament of what to do when software outlives its practitioners. *By Jon "maddog" Hall*

An article about the Voyager Deep Space Probes circulated recently on Facebook; after 40 years of flight, management was looking for programmers who knew Fortran, COBOL, and assembly language. The call for programmers proficient in these "ancient" languages garnered a lot of laughs from the younger programmers in our community but demonstrates a future problem that is just beginning to appear.

The one remaining original programmer on the Voyager project, Larry Zottarelli, is now 80 years old. Remember that Voyager has been flying for about 40 years, and its design really began perhaps five years before it was manufactured, tested, and launched, when Zottarelli was still fairly young. Probably a lot of the people working on Voyager did not think it would still be working 38 years later and would not believe that it might still be operating well into the 2020s. NASA needs programmers who can keep the software "alive."

This is not the first time I have run into situations of ancient hardware and software, but it is one of the most interesting because the hardware cannot be upgraded.

Many times, the solution is simply to develop a new system, run it in parallel with the old system to make sure they create the same answers, then discard the old system. This process can generate maintenance cost savings that pay for the redesign and re-implementation of the new system.

Another NASA project was the redesign of the Johnson Space Flight control center more than 20 years ago. The previous control center had been built more than 20 years before THAT, and its hardware was hopelessly out of date. Spare parts were hard to find, and each year the cost of maintenance went up. At the time I became involved, NASA was paying more than $200 million a year just on the maintenance of their ancient hardware and software.

A project was proposed to rewrite all the software and replace the hardware with a "standard operating system" and modern production hardware. The cost of this replacement, which gave much better reliability and flexibility than the old systems, would be (coincidentally) $200 million, the same as one year of maintenance on the old systems. However, the estimated yearly maintenance cost would "only" be $23 million and, more importantly, could be accomplished by younger engineers using more modern tools.

The use of these younger engineers was not (as some people commenting on the Voyager article thought) to save money by hiring younger, lower paid, instead of older, higher paid engineers, but to have the luxury of using engineers who were not on the verge of retiring or (worse yet) dying. It is one thing to bring back an engineer from retirement to fix something and quite another to try and bring them back from the dead.

The new system was designed and built using the Digital Unix operating system and DEC Alpha systems hardware. Two hundred million hours of telemetry data was fed through both the old and new systems to make sure the new system worked exactly the same as the old system.

A third "oldie" story has to do with automated teller machines (i.e., ATMs) used by banks. Many of these machines were designed in the days of the Intel x386 processors and had been functioning very well for a decade or more using IBM's OS/2 operating system. Then IBM dropped support for OS/2, and the ATMs were left without anyone to provide software maintenance.

At the time this happened, Microsoft had also dropped support for the x386 computers, moving on to x486 and Pentium systems. Even if Microsoft had still supported the base hardware, the ATM machines typically did not have much memory in them, so modern Microsoft Windows systems did not work either. A lot of these banks chose GNU/Linux, not only because it did the job, but because they knew they would always be able to fix the problems that occurred.

Of course, there was also the unforgettable campaigns around the "Y2K problem" [1], which has probably already been forgotten by some of the younger IT crowd, along with the corresponding "End of the Unix Epoch" [2] that will be happening in a few years.

The problems we have not encountered yet will be the most interesting. We are dealing with a relatively young industry by historical standards, and some of the aging issues are just beginning to show up.

Do not, however, laugh at the "ancient languages," my young friends. Someday your favorite language may also be old, deprecated, and ancient.

Except for C and GNU/Linux, of course. ■■■

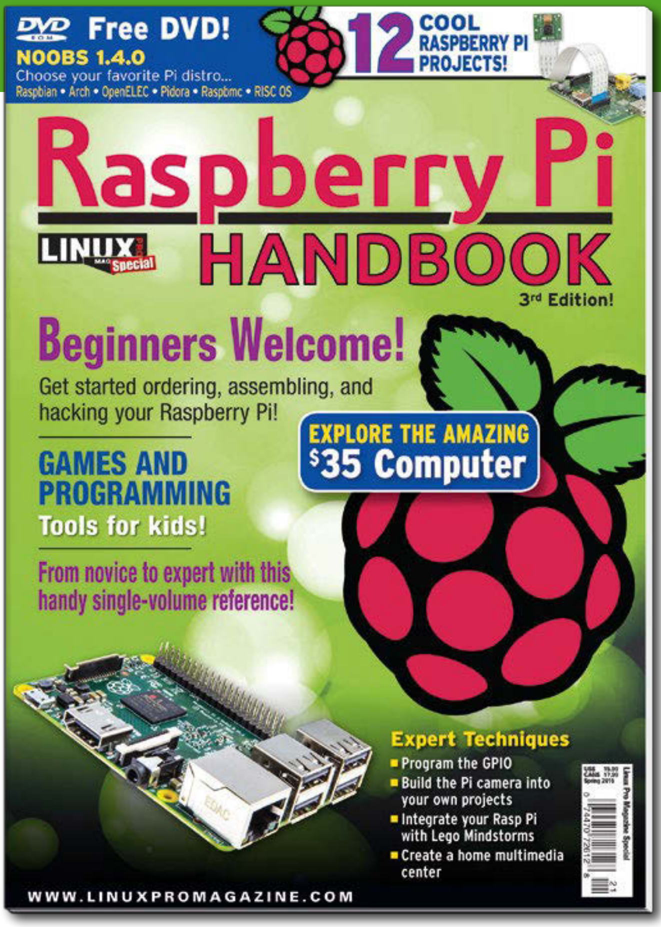
INFO

[1] Y2K problem: *https://en.wikipedia.org/wiki/Year_2000_problem*

[2] Year 2038 problem: *https://en.wikipedia.org/wiki/Year_2038_problem*

THE AUTHOR

Jon "maddog" Hall is an author, educator, computer scientist, and free software pioneer who has been a passionate advocate for Linux since 1994 when he first met Linus Torvalds and facilitated the port of Linux to a 64-bit system. He serves as president of Linux International®.



Free DVD! NOOBS 1.4.0 Choose your favorite Pi distro... Raspbian • Arch • OpenELEC • Pidora • Raspbmc • RISC OS

12 COOL RASPBERRY PI PROJECTS!

Raspberry Pi LINUX MAG Special HANDBOOK 3rd Edition!

Beginners Welcome! Get started ordering, assembling, and hacking your Raspberry Pi!

EXPLORE THE AMAZING $35 Computer

GAMES AND PROGRAMMING Tools for kids!

From novice to expert with this handy single-volume reference!

Expert Techniques
- Program the GPIO
- Build the Pi camera into your own projects
- Integrate your Rasp Pi with Lego Mindstorms
- Create a home multimedia center

WWW.LINUXPROMAGAZINE.COM

Crowd Supply boosts open hardware

Fan Funding

A small crowdfunding site is helping to boost the growth of open hardware businesses. *By Bruce Byfield*

Crowdfunding began as a way for amateurs to fund their projects. When professional projects like the *Veronica Mars* movie [1] started using it, many people complained. Today, however, the combination of crowdfunding and free software has made open hardware a reality and is starting to create a new niche of small businesses. Prominent among those promoting this trend is Crowd Supply [2], a crowdfunding site that carries the approval of the Free Software Foundation [3].

Josh Lifton, Crowd Supply's CEO, sees a natural connection between crowdfunding and free software. "Both rely on a distributed group of people," Lifton observes, "most of whom have never met. Participants in both often get their start because of their own interests, and both, too, are motivated by a degree of idealism," he continues. "[They] involve looking at the value of a product beyond its profit margin. It's more about how does this makes life better – and not end up as just another piece of landfill."

Crowd Supply was founded two and a half years ago by a collection of engineers and designers. Already developing a modern stenotype [4], Lifton and his team essentially created a site for the kinds of assistance that they would appreciate themselves. "We saw that people were starting to use Kickstarter and Indiegogo as product launch programs instead of just pure art patronage or charity," Lifton says. "We intentionally started Crowd Supply as a product launch platform" – a set of services and expertise that first-time product producers could use to increase their chances of success.

Besides its specific focus on free software and open hardware, what makes Crowd Supply unique is that it is not just a site on which to post crowdfunding campaigns. "We only do physical products," Lifton says. "We don't do software, and we don't do performance-based things. If you're not shipping something to your customer, then we won't do it. Everything else is built around that basic concept. We offer fulfillment services, we offer marketing, we add sales and e-commerce as one re-seller, and, of course, we help people."

The Rocky Road to Market

Before working with a client, Crowd Supply analyzes its business plan carefully. Lifton describes its initial analysis as "a gut check for people who have no idea how to bring a product to market. It's not a deep analysis," he concedes, "but no one else is doing it." What Crowd Supply is looking for is "the story to be told around [a product]," how much its developers want to raise, and how realistic their plans are.

Sometimes, Crowd Supply rejects ideas out of hand, such as perpetual motion machines, at least one of which was actually presented to the company. However, according to Lifton, most of the campaigns it rejects have failed to validate their business plans by presenting their products to any sort of test group or trying to sell a test batch. "We never reject [campaigns] outright," Lifton says. "We always give some sort of advice and suggest what can be done, so we will consider them again." Crowd Supply does not get re-applications very often, "but it has happened."

The clients for whom Crowd Supply is likely to work are usually "professionals who can design a product, but who don't have the backing of a logistics framework to do accounting, source components, or things like that." Often the clients they accept are engineers, and often they are working on their first product.

After working with more than a hundred campaigns, Crowd Supply has developed a strong sense of which are likely to succeed. "It doesn't take us very long to assess where a product is when it comes through the door," Lifton says. He has noticed two common trends among those whose campaigns succeed. First, they are usually "personally vested" in their product – not financially, Lifton is quick to explain, but in the sense that "it's their passion." Second, "they are catering towards a very well defined niche audience. You might think of the better mousetrap and think, 'well, everyone would want a new mousetrap', but in fact it's hard to market to everyone. Instead, you should concentrate on a niche audience that appeals to people who want to put their money where their mouth is. They may not have a lot of money, so I wouldn't consider them luxury consumers. I would instead consider them value-based consumers. Their values make them wait for the laptop or bicycle that matches their values."

Outside of open hardware, an example of such a niche product is the Portland Press [5], a high-end coffee maker. As Lifton points out, people can buy a product with the same functionality at Kmart for one-tenth the price, but the Portland Press appeals to customers who appreciate quality. Similarly, inside open hardware, Librem [6], the free-software laptop that is probably Crowd Supply's best-known campaign, appeals to those who support the Free Software Foundation's ideals of consumer control, although Lifton suggests that it could eventually reach a much larger market if concerns about security and privacy continue.

Crowd Supply does not take equity in the campaigns it agrees to work with, nor does it manufacture anything itself, although it does pass along information about manufacturers that successful clients have worked with.

It also gives advice in all the steps on the way to market, from manufacturing methods, to validating business plans via the initial campaign, to producing a second manufacturing run after the campaign. For example, instead of housing hardware in an injection-molded case, it might suggest reducing cost and adding value by building a hand-made case instead. Essentially, Crowd Supply fills in the gaps that would otherwise require more employees.

After the crowdfunding campaign, Crowd Supply might also offer advice about how to get a product into brick and mortar or online stores. It also retains the right to sell products on its own site, which provides much of its revenue. "We only make money if your campaign is successful." Lifton says.

Throughout this process, Crowd Supply also tries to manage clients' expectations. "It's a huge leap to go from idea to prototype, another order of magnitude leap to go to production, and another order of magnitude to go from your first run to steady availability. And none of these things are particularly stable; it's easy to slide back. That's why many businesses fail." At Crowd Supply, Lifton says, "we try to prepare people the best we can for each of those steps."

The process is rigorous, but it seems to pay off. "The one thing that really sets us apart from everyone else is that one hundred percent of those we have funded delivered a product to their customers," Lifton claims. Lifton also states that Crowd Supply's open hardware campaigns have a 56 percent success rate – twice that of Kickstarter's and several times that of Indiegogo's.

Growing Pains

Lifton concedes that small businesses that sell open hardware face problems. In particular, preparing a product for market can be expensive, especially in the small quantities that most crowdfunding campaigns involve. "Maintaining the supply chain is a constant struggle," Lifton says, and sometimes ingenious alternatives need to be found.

Similarly, new manufacturers can have trouble finding vendors willing to take a chance on them. For example, the Portland Press is now being carried by Amazon and Starbucks, as well as other boutique shops, but the Novena [7], a laptop that was an early success, is currently being carried only by Crowd Supply.

Still, Lifton suggests that the open hardware niche continues to expand, because it fills a need. Crowdfunding, he says, "is a way for people who would otherwise fall between the cracks of venture capital or bootstrapping to bring products to life." Speaking of Librem, he adds, "I don't think they could have gotten venture funding – not that I think they necessarily wanted to. The market for software freedom is now taking form in the minds of people who never cared about it before – not as freedom necessarily, but as security and privacy, and [as a way of] breaking the bonds of corporate ownership."

In fact, Lifton says that "it's not too long before alternatives to all those big companies will emerge. I think we have some of them already that have launched." Yet even if that prediction turns out in a few years to be overoptimistic, one thing seems clear: Thanks to pioneers like Crowd Supply, open hardware is no longer a fantasy, but an increasingly plausible approach for small business. ■■■

INFO

[1] *Veronica Mars* movie: *https://www.kickstarter.com/projects/559914737/the-veronica-mars-movie-project*

[2] Crowd Supply: *https://www.crowdsupply.com*

[3] "Founder of GNU bestows blessing upon open hardware-focused crowdfunding site," by Sean Gallagher: *http://arstechnica.com/information-technology/2015/07/founder-of-gnu-bestows-blessing-upon-open-source-crowdfunding-site/*

[4] Next gen open stenotype: *https://www.crowdsupply.com/open-steno-project/next-generation-open-stenotype*

[5] Portland Press: *https://www.crowdsupply.com/bucket/the-portland-press*

[6] Librem 13: *https://www.crowdsupply.com/purism/librem-13*

[7] Novena: *https://www.crowdsupply.com/sutajio-kosagi/novena*

Zack's Kernel News

Chronicler Zack Brown reports on the latest news, views, dilemmas, and developments within the Linux kernel community.

By Zack Brown

Compiling the Kernel as a Library

Octavian Purdila came up with a way to compile the kernel as a static library, called LKL (Linux Kernel Library), making all of its interfaces available to software running on other operating systems. The goal, Octavian said, was "to allow reusing the Linux kernel code as extensively as possible with minimal effort and reduced maintenance overhead."

Octavian distinguished LKL from UML (User Mode Linux), pointing out that UML offered a full operating system environment, with corresponding infrastructure requirements like filesystems and processes, whereas LKL is a programming library with a set of function APIs that any program could link to and use.

Richard Weinberger said that this librarification "eliminates UML's most problematic areas, system call handling via `ptrace()` and virtual memory management via `SIGSEGV`."

Richard asked whether LKL was currently restricted to single threading only, and Octavian replied, "at this point yes. SMP support is on my todo list though."

Several folks jumped into the discussion, mostly regarding compatibility with similar projects such as libOS and libguestFS. These aren't necessarily the sort of projects that require acceptance by Linus Torvalds or any of the other top kernel contributors. There's a certain amount of access to users that comes when a project has a dedicated build target within the kernel or its own driver or filesystem, but for projects like these, that aim mostly for specialized use cases, it's often sufficient to keep them as standalone projects.

Turning Off Portions of a Device to Save Power

Irina Tirdea posted some patches to allow the kernel to suspend a piece of hardware attached to the system (e.g., turning off video in response to closing the lid of a laptop or the screen of a phone). Currently, the SysFS control power interface has only two options: *on* and *auto*. Irina wanted to add a new option: *off*. As she described it, "the device will be force suspended by calling its runtime suspend callback and disabling runtime power management so that further accesses to the device will not change the actual state. The device can be resumed by setting the attribute to *on* or *auto*."

Rafael J. Wysocki dismissed the whole idea, saying, "Had we thought this had been a good idea, we'd have added that thing to the interface from the start."

The reason, he said, was that userspace generally had no way of knowing when it was safe to suspend a device. Putting that kind of control into user software, he said, could break things.

Irina pointed out that in the scenarios she'd mentioned, it wasn't software but the user who initiated the suspend, by closing the lid of the device or interacting with it in some other physical way. She said that in the current code, drivers for touchscreens and other hardware each had to have their own mechanism for suspending when not needed. She said, "This adds more complexity to every driver by adding one more logical power state. It would be good to have a common interface instead of doing this in every driver."

Oliver Neukum also had doubts about Irina's approach. It seemed to him that the software could accomplish the same thing by simply stopping using a piece of hardware, thus letting that hardware idle and use less power. Why bother suspending at all? In Irina's approach, he said, there were various complex issues, including the need to monitor all the software lock counts, to make sure the hardware was truly free to suspend.

Octavian Purdila offered some clarification of Irina's work. He said:

The very specific problem we want to solve is handling touchscreens on a phone/tablet. When the screen is turned off, it is ideal to suspend the touchscreen for two reasons: to lower the power consumption as much as possible and to prevent interrupts to wake-up the CPU when the user touches the device, and thus save even more power as we allow the CPU to stay in deep idle states for longer periods.

Note that when the screen is turned on again, we want to resume the touchscreen so that it can send events again.

This is different than the lid closes examples, as in that case the user can not generate new events and thus the usual autosuspend feature is probably good enough (if the suspend power and autosuspend power consumption is similar).

ZACK BROWN

The Linux kernel mailing list comprises the core of Linux development activities. Traffic volumes are immense, often reaching 10,000 messages in a week, and keeping up to date with the entire scope of development is a virtually impossible task for one person. One of the few brave souls to take on this task is **Zack Brown**.

Rafael said he and Alan Stern had discussed something a few months back that might address Octavian's example better. As he described it, it should be possible "to add a third value to `/sys/devices/` ... `/power/wakeup` (in addition to 'disabled' and 'enabled') so userspace can indicate that remote wakeup should not be enabled for runtime suspend for the device (since there's no way to indicate that today)."

Alan, however, added a caveat for this idea, saying that "it was never implemented. For that reason, it was never completely fleshed out."

Several folks debated the issue, but the discussion seemed focused more on implementation than on whether Irina's feature would be good or not. Irina's code seemed to be generally disapproved of, but there was no consensus on what would be better. It was difficult even to identify the specific use cases that any proposal might address. For example, at one point in the discussion Dmitry Torokhov said:

In ChromeOS, we have a custom 'inhibit' control that:

1. Tells input core to ignore all events from a given device

2. Allows driver to put device in low power mode if driver desires to do so. The driver can do it via runtime PM or on its own. Usually on its own since when using runtime PM userspace may disable it, which may not be desirable.

I would love to have something generic instead of input-specific.

But, he added, "I was hesitant bringing it upstream as I believe it is not necessarily input device specific, and I would love to have it implemented at device core level."

Ultimately, Rafael and Alan were most active in trying to come up with an appropriate approach, but no solid design emerged from the discussion.

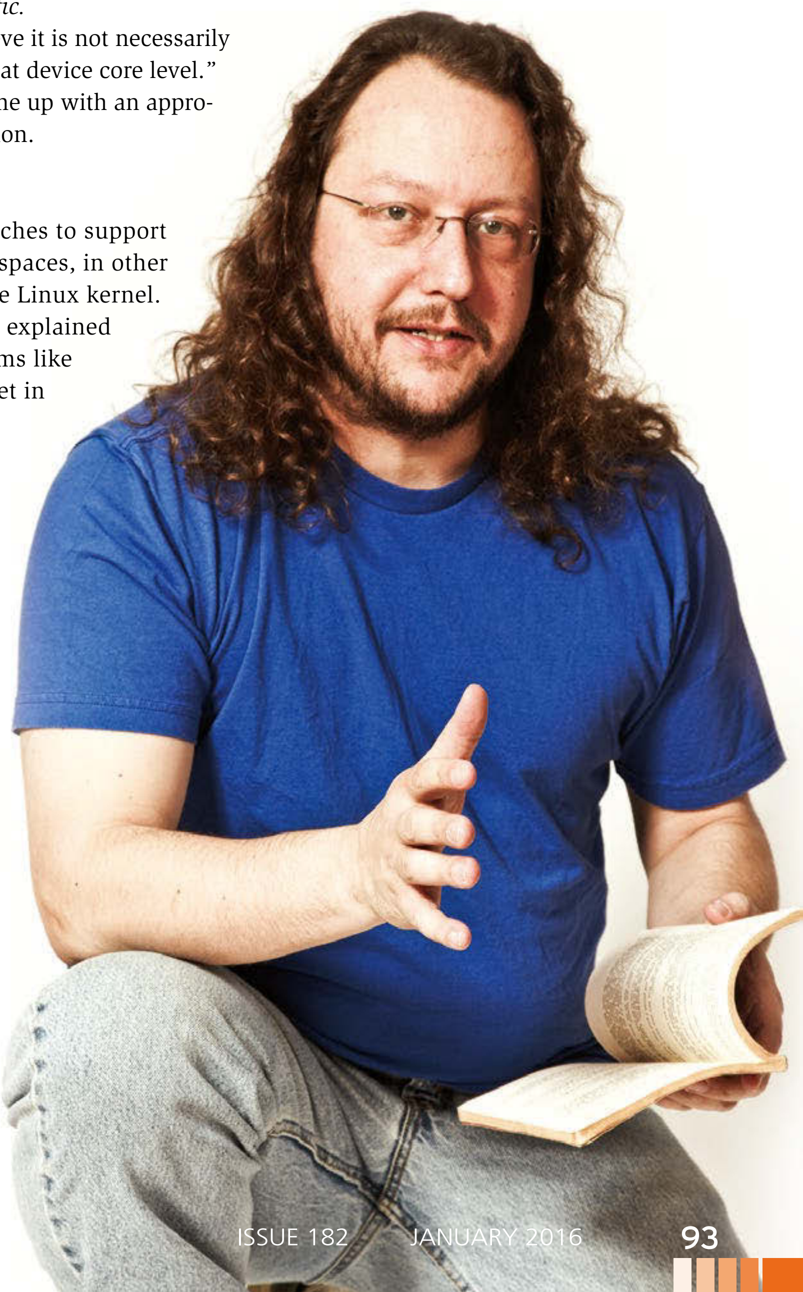
Mounting Filesystems Under Emulation

Seth Forshee and Eric Biederman were working on some patches to support mounting ext4 and FUSE filesystems from within user namespaces, in other words from within an emulated system running on top of the Linux kernel. Seth posted an initial set of patches for consideration. As he explained in a follow-up email, "This is supporting mounting filesystems like ext4 by unprivileged users and not trusting the labels they set in the same way as we trust labels on filesystems mounted by privileged users."

Note that "labels" in this context does not refer to filesystem labels that can be used to determine target mountpoints for a given filesystem. Instead, "labels" here refers to a set of extended attributes (`xattr`) data used by the Linux Security Module (LSM) to constrain user access to a given filesystem.

Casey Schaufler pointed out a potential conflict with another bit of coding being done by Lukasz Pawelczyk, to support LSMs in user namespaces. He said that Seth and Eric's work "gives an unprivileged user the ability to ignore the Smack labels that are on files and to create files with labels that do not match the rules laid down by the security module."

Casey thought that ignoring the Smack labels would leave security holes, allowing untrusted users to access files that would otherwise be protected. He said, "you can't pick and choose when you are going to pay attention to the security attributes on a filesystem. It's possible that it will work out the way you want it, but it probably won't. Smack doesn't allow you to choose if you're using xattrs. SELinux does, but certainly doesn't expect you to be flipping it on and off.



I'm not convinced that it's safe to do for capability sets, either."

Andy Lutomirski felt that Casey's concerns might not apply in the current situation. He suggested that, "If I mount an unprivileged filesystem, then either the contents were put there *by me*, in which case letting me access them are fine, or (with Seth's patches and then some) I control the backing store, in which case I can do whatever I want regardless of what LSM thinks."

Casey replied, "If you have a security module that uses attributes on the filesystem you can't ignore them just because it's 'your data'. Mandatory access control schemes, including Smack and SELinux don't give a fig about who you are. It's the label on the data and the process that matter. If 'you' get to muck the labels up, you've broken the mandatory access control."

Eric replied:

There are two fundamental issues mounting filesystems without privilege, by which I actually mean mounting filesystems as the root user in a user namespace.

- Are the semantics safe?

- Is the extra attack surface a problem?

Figuring out how to make semantics safe is what we are talking about.

Once we sort out the semantics we can look at the handful of filesystems like fuse where the extra attack surface is not a concern.

With that said, desktop environments have for a long time been automatically mounting whichever filesystem you place in your computer, so in practice what this is really about is trying to align the kernel with how people use filesystems.

I haven't looked closely, but I think docker is just about as bad as those desktop environments when it comes to mounting filesystems.

Eric also added:

There are filesystems like fat and minix that can not store a label. Since it is not possible to store labels securely in filesystems mounted by unprivileged users (at least in the normal sense), the intent would be to treat a filesystem mounted without the privileges of the global root user as a filesystem that does not support xattrs.

Treating such a filesystem as a filesystem that does not support xattrs is the only possible way support such a filesystem securely, because as you have said someone who can muck up the labels breaks mandatory access control.

Given how non-trivial it is to grasp the nuances of different lsms mandatory access control semantics, I am asking Seth for the first [pass] to simply forbid mounting of filesystems with just user namespace permissions when there is an lsm active.

Once we get that far, smack may never need to support such systems.

Meanwhile, Lukasz also took a look at Seth and Eric's work and thought that the conflict Casey had mentioned earlier was not a real problem. He said, "If your approach here is to treat user ns mounted filesystems as if they didn't support xattrs at all, then my patches don't conflict here any more than Smack itself already does." Seth said he'd make sure to check out Lukasz's patches in any case and make sure there were no issues.

At this point, the conversation descended into a consideration of specific user scenarios that might or might not expose private data to an untrusted user. Various folks joined in, trying to identify exactly where in the code security would break down and what needed to happen at those points to firm it up. At one point, while working through one of these scenarios, Casey remarked, "My position is that there's a workaround but that the design is still fundamentally flawed."

At a different point in the conversation, Seth said, "Right now, it looks to me like the only safe thing to do with mounts from unprivileged users is to ignore the security labels, so that's what I'm trying to do with these changes. If there's some better thing to do, or some better way to do it, I'm more than happy to receive that feedback." Casey replied, "Personally, I don't believe that the goal of supporting unprivileged mounts is especially sane. I am willing to be educated, but I don't see a rational solution."

There was ultimately no resolution to the various disagreements. The issues are very thorny to work through, as are all kernel features related to security. Sometimes the only solution is to support a subset of features that appears arbitrary to the end user but is absolutely required for security. ■■■

FEATURED EVENTS

Users, developers, and vendors meet at Linux events around the world. We at *Linux Magazine* are proud to sponsor the Featured Events shown here.

For other events near you, check our extensive events calendar online at *http://linux-magazine.com/events.*

If you know of another Linux event you would like us to add to our calendar, please send a message with all the details to *events@linux-magazine.com*.

Embedded Linux Conference

Date: April 4–6, 2016

Location: San Diego, California

Website: *http://events.linuxfoundation.org/events/embedded-linux-conference/*

ELC, the technical conference for companies and developers using Linux in embedded products, presents the 12th year of sessions dedicated exclusively to embedded Linux and embedded Linux developers.

Apache: Big Data

Date: May 9–11, 2016

Location: Vancouver, British Columbia

Website: *http://events.linuxfoundation.org/events/apache-big-data-north-america*

Apache projects are the foundation of many Big Data platforms. Join other professionals working in Big Data, ubiquitous computing, and data engineering and science to accelerate the state of the art.

ApacheCon North America

Date: May 11–13, 2016

Location: Vancouver, British Columbia

Website: *http://events.linuxfoundation.org/events/apachecon-core-north-america*

Join the open source community to learn about and collaborate on the technologies and projects driving the future of open source, web technologies, and cloud computing.

EVENTS

Fosdem '16	January 31–31	Brussels, Belgium	https://fosdem.org/2016/
FAST '16	February 22–25	Santa Clara, California	https://www.usenix.org/conference/fast16
Icinga Camp	March 1	Berlin, Germany	https://www.icinga.org/community/events/icinga-camp-berlin/
NSDI '16	March 16–18	Santa Clara, California	https://www.usenix.org/conference/nsdi16
Collaboration Summit	March 29–31	Lake Tahoe, California	http://events.linuxfoundation.org/events/collaboration-summit
Embedded Linux Conference	April 4–6	San Diego, California	http://events.linuxfoundation.org/events/embedded-linux-conference
Linux Storage Filesystem and MM Summit	April 18–19	Raleigh, North Carolina	http://events.linuxfoundation.org/events/linux-storage-filesystem-and-mm-summit
Vault Linux Storage and Filesystems Conference	April 20–21	Raleigh, North Carolina	http://events.linuxfoundation.org/events/vault
Open Source Data Center Conference	April 26–28	Berlin, Germany	https://www.netways.de/en/events_trainings/osdc/overview/
Apache Big Data North America	May 9–11	Vancouver, BC, Canada	http://events.linuxfoundation.org/events/apache-big-data-north-america
DrupalCon North America	May 9–13	New Orleans, Louisiana	https://events.drupal.org/neworleans2016
ApacheCon Core North America	May 11–13	Vancouver, BC, Canada	http://events.linuxfoundation.org/events/apachecon-core-north-america
USENIX ATC '16	June 22–24	Denver, Colorado	https://www.usenix.org/conference/atc16

Images © Alex White, 123RF.com

CALL FOR PAPERS

We are always looking for good articles on Linux and the tools of the Linux environment. Although we will consider any topic, the following themes are of special interest:

- System administration
- Useful tips and tools
- Security, both news and techniques
- Product reviews, especially from real-world experience
- Community news and projects

If you have an idea, send a proposal with an outline, an estimate of the length, a description of your background, and contact information to *edit@linux-magazine.com*.

The technical level of the article should be consistent with what you normally read in *Linux Magazine*. Remember that *Linux Magazine* is read in many countries, and your article may be translated into one of our sister publications. Therefore, it is best to avoid using slang and idioms that might not be understood by all readers.

Be careful when referring to dates or events in the future. Many weeks could pass between your manuscript submission and the final copy reaching the reader's hands. When submitting proposals or manuscripts, please use a subject line in your email message that helps us identify your message as an article proposal. Screenshots and other supporting materials are always welcome.

Additional information is available at:
http://www.linux-magazine.com/contact/write_for_us.

AUTHORS

Erik Bärwaldt	42
Chris Binnie	50
Zack Brown	92
Bruce Byfield	76, 90
Joe Casad	3, 8
Tobias Eggendorfer	30
Jon "maddog" Hall	88
Charly Kühnast	48
Hartmut Noack	64
Dmitri Popov	70
Dr. Holger Reibold	80
Thorsten Scherf	24
Mike Schilli	60
Tim Schürmann	36
Mark Vogelsberger	54
Uwe Vollbracht	20
Harald Zisler	12

CONTACT INFO

Editor in Chief
Joe Casad, jcasad@linux-magazine.com

Managing Editor
Rita L Sooby, rsooby@linux-magazine.com

Localization & Translation
Ian Travis

News Editor
Joe Casad

Copy Editor
Amber Ankerholz

Layout
Dena Friesen, Lori White

Cover Design
Dena Friesen, Lori White

Cover Image
© Zoya Fedorova, 123RF.com

Advertising – North America
Ann Jesse, ajesse@linuxnewmedia.com
phone +1 785 841 8834

Advertising – Europe
Penny Wilby, pwilby@sparkhausmedia.com
phone +44 1807 211100

Publisher
Brian Osborn, bosborn@linuxnewmedia.com

Marketing Communications
Darrah Buren, dburen@linuxnewmedia.com
Linux New Media USA, LLC
616 Kentucky St.
Lawrence, KS 66044 USA

Customer Service / Subscription
For USA and Canada:
Email: cs@linuxpromagazine.com
Phone: 1-866-247-2802
(Toll Free from the US and Canada)
Fax: 1-785-856-3084

For all other countries:
Email: subs@linux-magazine.com
Phone: +49 89 99 34 1167
Fax: +49 89 99 34 1198

www.linuxpromagazine.com – North America

www.linux-magazine.com – Worldwide

While every care has been taken in the content of the magazine, the publishers cannot be held responsible for the accuracy of the information contained within it or any consequences arising from the use of it. The use of the disc provided with the magazine or any material provided on it is at your own risk.

Printed in Germany

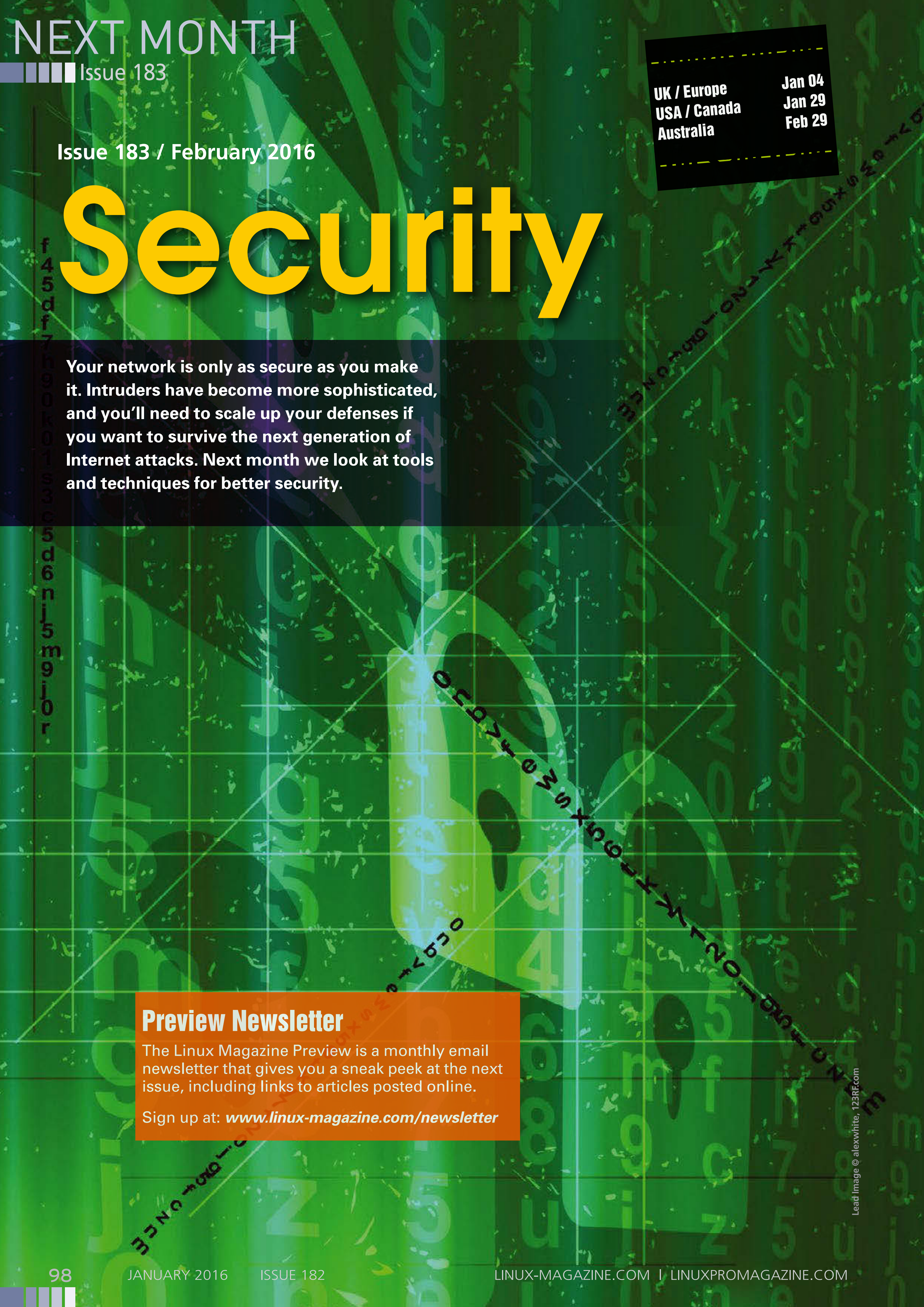
Distributed by COMAG Specialist, Tavistock Road, West Drayton, Middlesex, UB7 7QE, United Kingdom

Published in Europe by: Sparkhaus Media GmbH, Putzbrunner Str. 71, 81749 Munich, Germany.

NEXT MONTH

Issue 183

UK / Europe	Jan 04
USA / Canada	Jan 29
Australia	Feb 29



Issue 183 / February 2016

Security

Your network is only as secure as you make it. Intruders have become more sophisticated, and you'll need to scale up your defenses if you want to survive the next generation of Internet attacks. Next month we look at tools and techniques for better security.

Preview Newsletter

The Linux Magazine Preview is a monthly email newsletter that gives you a sneak peek at the next issue, including links to articles posted online.

Sign up at: ***www.linux-magazine.com/newsletter***

Lead Image © alexwhite, 123RF.com

MOBILE USERS

search for us today at your digital newsstand!

Only a swipe away!

Download our convenient digital editions for your iPad, iPhone, or Android device.

GOOGLE PLAY MAGAZINES

ADMIN Magazine

Linux Pro Magazine

Ubuntu User

Raspberry Pi Geek

APPLE NEWSSTAND

ADMIN Magazine

Linux Pro Magazine

Ubuntu User

Raspberry Pi Geek

Visit our apps page for more information: www.medialinx-shop.com/apps