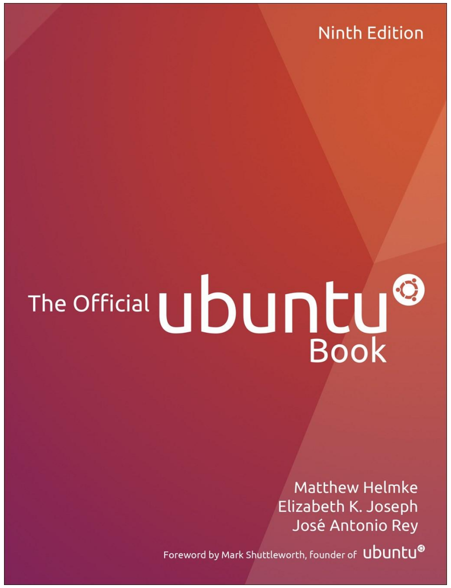
Keeping Your Ubuntu Systems Secure

UbuCon Summit @ SCaLE16x
8 March 2018

Elizabeth K. Joseph
lyz@ubuntu.com
@pleia2



Elizabeth K. Joseph

- Ubuntu Member since 2007
- Co-author, The Official Ubuntu Book, 8th and 9th editions
- Systems Administrator by trade, working as a Developer Advocate at Mesosphere

What to expect

Hint: Probably no big revelations ;)

Security basics

Default security features

Understanding your system

Managing software

Ubuntu on servers

Security Basics

Install all security updates

Use a supported version of Ubuntu

Use trusted software sources

Don’t run commands you don’t understand

Use firewalls, separate users, and disk encryption

Default security features

CC BY 2.0: https://www.flickr.com/photos/pleia2/26747405158/

Separate users

User isolation, both from each other and from administrative tasks, is a core security feature in Linux.

Every Linux system comes with a “root” user, which is only used for administrative tasks.

In Ubuntu, using this user directly is discouraged, instead preferring use of “sudo” by individual “administrator” users.

While regular desktop users cannot install new packages, a non-administrative account is perfectly functional for day to day use.

Tip: For added security, even as the owner (or only user on a system) you may want to create a separate user for doing administrative tasks, and do your day to day work on a user that does not have sudo access.

Debian & Ubuntu security teams

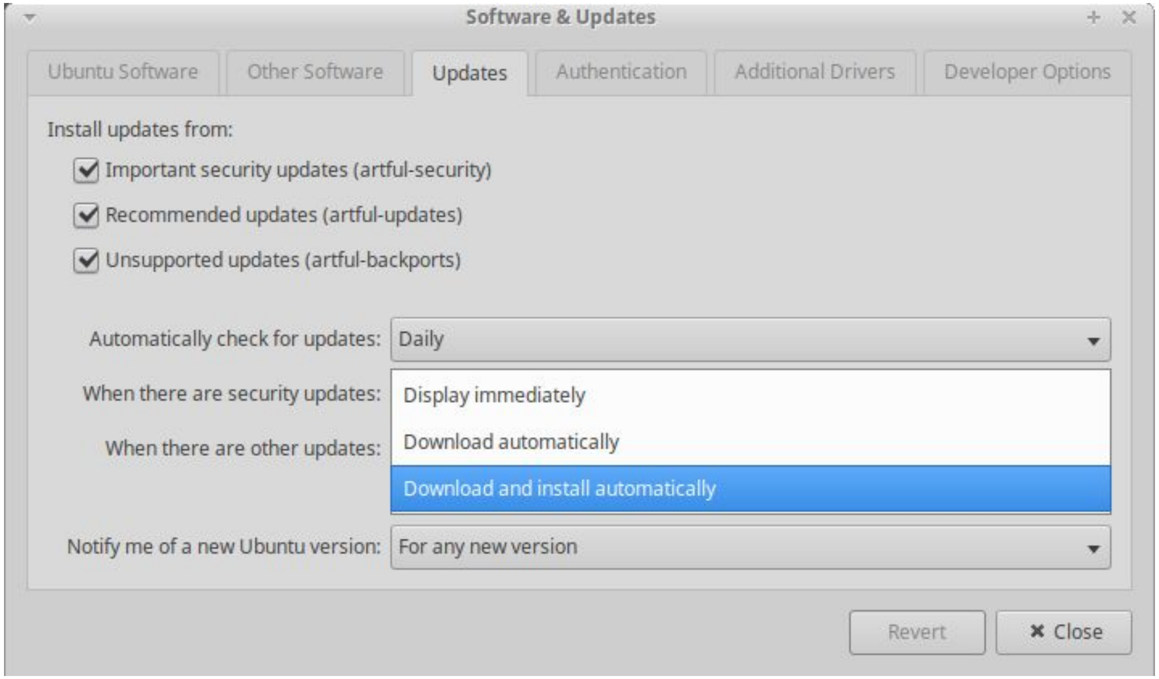
"Debian takes security very seriously. We handle all security problems brought to our attention and ensure that they are corrected within a reasonable timeframe." via https://www.debian.org/security/

All packages in Debian and Ubuntu "main" (and "restricted" in Ubuntu) are overseen by teams of security experts from Debian, Ubuntu and Canonical. All supported releases get attention to security vulnerabilities of all these packages.

You may subscribe to the ubuntu-security-announce mailing list to be notified when security updates are released: https://lists.ubuntu.com/mailman/listinfo/ubuntu-security-announce

Automatic security updates

If you’re looking for an easy way to keep a system up to date with security updates, you can enable automatic upgrades for security.



Install updates from:

☑ Important security updates (artful-security)

☑ Recommended updates (artful-updates)

☑ Unsupported updates (artful-backports)

Automatically check for updates: Daily

When there are security updates:

When there are other updates:

Display immediately

Download automatically

Download and install automatically

Notify me of a new Ubuntu version: For any new version

Hard drive encryption

During installation of Ubuntu you are presented with the option to encrypt your hard drive.

This helps keep your system secure, even when someone gains physical access to your machine, otherwise you have the following common attack vectors:

- Use a LiveCD/USB stick to boot/reboot your machine and:
  - Mount the hard drive, with full access to your data
  - Change or add users and passwords
  - Make any other changes they wish, including installing new software, without you knowing
- Physically remove your hard drive and mount it on another computer

Uncomplicated Firewall (ufw)

"The Uncomplicated Firewall (ufw) is a frontend for iptables and is particularly well-suited for host-based firewalls. ufw provides a framework for managing netfilter, as well as a command-line interface for manipulating the firewall." via https://wiki.ubuntu.com/UncomplicatedFirewall

Available with Ubuntu by default since 8.04, you just need to enable it.

There's a graphical interface, but using the command line is also pretty simple.

Example: Enable ufw, allow ssh access, enable logging, and check the status of the firewall:

```
$ sudo ufw status
Status: inactive
$ sudo ufw allow ssh/tcp
$ sudo ufw logging on
$ sudo ufw enable
$ sudo ufw status
Status: active
```

AppArmor

“AppArmor is a Mandatory Access Control (MAC) system which is a kernel (LSM) enhancement to confine programs to a limited set of resources.” via https://wiki.ubuntu.com/AppArmor

Profiles are created on a per-application basis to control how that program behaves on your system, including:

- What files can read and edit on your system
- What libraries it can load
- What applications it can launch
- Lower level access, such as: signal, ptrace, sockets

AppArmor was first introduced in Ubuntu 7.04, and was turned on by default in Ubuntu 7.10.

The Ubuntu Community

While you shouldn’t implicitly trust all advice you get from strangers on the internet, the Ubuntu community is one of the strongest among Linux communities for support.

Support outlets include:

- Internet Relay Chat (IRC): https://help.ubuntu.com/community/InternetRelayChat
- Mailing lists: https://lists.ubuntu.com/
- StackExchange: https://askubuntu.com/
- Forums: https://ubuntuforums.org/

The Ubuntu California team can also help you out in person :) https://ubuntu-california.org/

There’s a monthly meetup right here in Pasadena!

Understanding your system

CC BY 2.0: https://www.flickr.com/photos/pleia2/12601095294/

Linux gives you power

You can cause your computer to become completely non-functional with a single command.

Perhaps we all know about the one that deletes your entire system (or as much as it can get to before becoming non-functional)*:

```
$ rm -rf /
```

But making changes to everything from system binaries and libraries to your package repositories can make your system unusable.

The first step to understanding your system is understanding how it's organized.

* actually, this one is so famous Ubuntu won't allow you to do it anymore, and will throw an error:
rm: it is dangerous to operate recursively on '/'

Basic filesystem tour

The Linux file system is governed by the Filesystem Hierarchy Standard developed by the Linux Foundation.

- /home - individual user data
- /etc - system-wide configuration files
- /lib - system libraries
- /bin - system binaries (commands, executables, applications)
- /sbin - administrative binaries
- /usr - shared userland binaries, documentation, libraries
- /var - variable data such as logs, mail and print spools, etc
- ...and more! Including /boot /root /srv...

Learn more at: https://www.tldp.org/LDP/Linux-Filesystem-Hierarchy/html/

When you install a traditional Ubuntu package, it may install to several of these.

What is an Ubuntu (.deb) package?

An Ubuntu (or Debian) .deb file is an archive which includes:

1. A piece of software (application, library, kernel, etc) you wish to install.
2. The instructions for installing it system-wide, including dependencies and conflicts and the installation location for all files

Ubuntu also keeps track of packages being installed through a package management system which tracks .deb packages installed system-wide.

To install and track packages system-wide, root (administrative) permissions are required.

Simple package example

My first package was written for Debian, it was a single Perl script and corresponding configuration file and documentation.

Let’s have a look at **debian/install** for the first package I made:

```
dglog.pl usr/lib/cgi-bin
dglog.conf etc/
```

This instructed the package to install dglog.pl into /usr/lib/cgi-bin and dglog.conf to etc/

A **debian/rules** file gives further instructions, including installation of system-wide documentation for reference via browsing /usr/share/doc and via man pages. A **debian/control** file is what provides the package summary, dependencies and more.

Extract a .deb package without installing it some time and play around with it!

File permissions

Permissions on files in Linux can seem a bit cryptic at first, but they're important to understand and pretty easy to understand with a little arithmetic.

Let's look at a common example, your private SSH key:

```
$ ls -l .ssh/id_rsa
-rw------- 1 elizabeth elizabeth 1675 Jun 23  2014 .ssh/id_rsa
```

Only readable and writable by the user that owns them.

And then at configuration for your SSH server:

```
$ ls -l /etc/ssh/sshd_config
-rw-r--r-- 1 root root 2504 Jul 18  2016 /etc/ssh/sshd_config
```

Everyone on the system can read this configuration file, only the owner (root) can write to it.

File permissions continued

u = user who owns the file
g = group
o = others (not in u or g)

r = read, value 4
w = write, value 2
x = execute, value 1

So, what do these mean?

```
$ chmod g+w file.txt

$ chmod o-x file.txt

$ chmod 777 file.txt
```

Tip: Executable permissions are required for applications, but also for directories.

You won't be able to descend into a directory unless your user or group has +x privileges.

Something broken? Examine your commands

When something goes wrong with Ubuntu, we all head for our favorite search engine or support forum to learn how to fix it.

Be careful.

Never run commands you found online without understanding what they do.

Read the documentation for commands you're running, or simply ask for more help if you're unsure.

If understanding the fix truly eludes you, don't try it.

Managing software

CC BY 2.0: https://www.flickr.com/photos/pleia2/40267996371/

Never.

The answer to when it’s OK to use an unsupported version of Ubuntu.

Use supported versions

It’s never safe from a security perspective to use an unsupported version of Ubuntu.

Even when an unsupported version of Ubuntu “works better” on your system.

Even if you keep it firewalled off.

Even if you don’t connect it to the internet.

Maybe if you never turn it on again.

What versions are supported?

Ubuntu versions are date-based:

Year. Month
17.10
2017 October

Long-term support (LTS) releases are the first (of two) release of the year on even years, and are supported for **5 years***.

All other releases are supported for **9 months**.

Get Release and End of Life emails by subscribing to the ubuntu-annouce mailing list:
https://lists.ubuntu.com/mailman/listinfo/ubuntu-announce

* Tip: Not all flavors are created equal. Xubuntu, for instance, only supports a 3 year LTS.

Ubuntu Released/Supported party* game!

- 16.04
  - Released in 2016, in April.
  - An LTS!
  - Supported until April 2021.
- 16.10
  - Released in 2016, in October.
  - Was supported until July 2017.
- 17.04
  - Released in 2017, in April.
  - Was supported until January 2018.
- 17.10
  - Released in 2017, in October.
  - Supported until July 2018.
- 18.04
  - Will be released in 2018, in April.
  - An LTS!
  - Supported until April 2023.

* you probably don’t want to invite me to parties

Ubuntu packages

In your /etc/apt/sources.list file you may have a series of line that look like this:

```
deb http://security.ubuntu.com/ubuntu artful-security main restricted
deb http://security.ubuntu.com/ubuntu artful-security universe
deb http://security.ubuntu.com/ubuntu artful-security multiverse
```

*“All binary packages in **main** and **restricted** are supported by the Ubuntu Security team for the life of an Ubuntu release,*

*while binary packages in **universe** and **multiverse** are supported by the Ubuntu community.”*

via https://wiki.ubuntu.com/SecurityTeam/FAQ (formatting added)

Personal Package Archives (PPAs)

Frequently referred to simply as “PPAs” the Personal Package Archives are .deb packages hosted on Launchpad.net and make it easier for developers to release software for Ubuntu.

Beware!

1. These live outside of standard Ubuntu release and security processes.
2. There are absolutely no guarantees from Canonical or the Ubuntu community that these will work, or that they are safe.
3. Anyone can create a PPA!

Remember: Package installs are done as the root user. A malicious user who created a PPA now has administrative rights on your computer. They can destroy your system, install software you didn’t intend to, add a back door so they can control remotely it, and anything else they want.

If you use a PPA, make sure you can trust the creator.

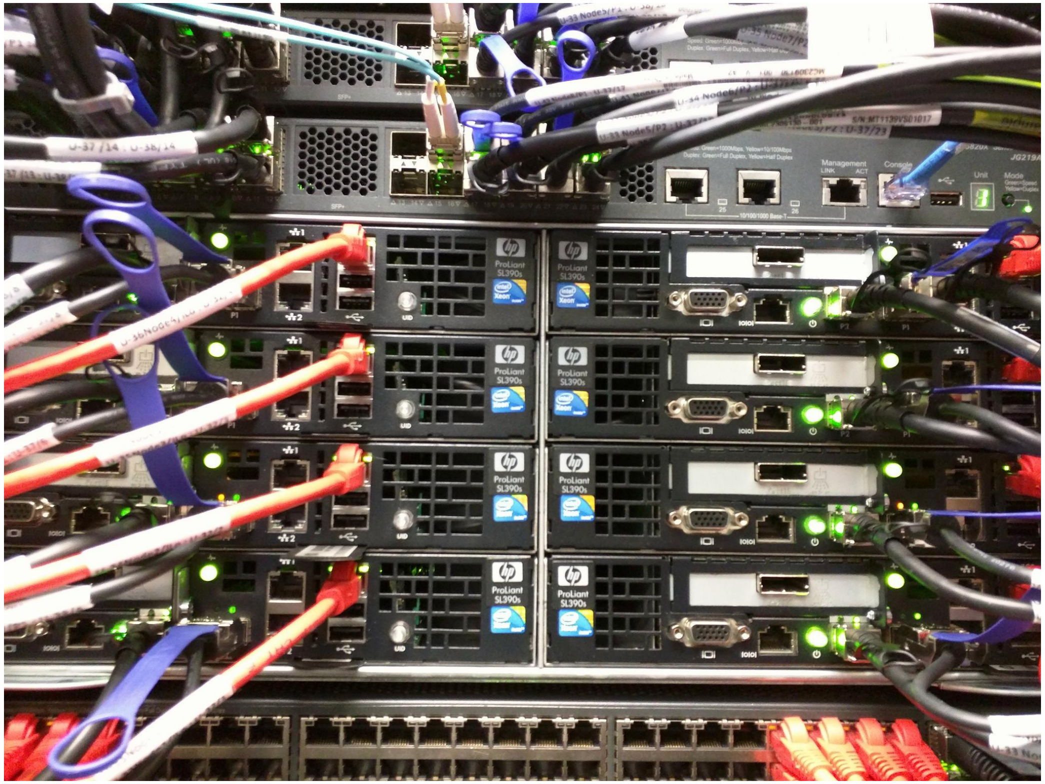
Other software

It may be common for you to download software from the internet or buy it off a shelf and run it.

This is less common in Ubuntu because you have a huge repository of packages at your fingertips, you may not need anything else!

But sometimes you do. **Be cautious.** Consider the following:

- If you're using "sudo" or the root user, remember it has access to your entire system.
- The software has not been evaluated by the Ubuntu security team, you need to trust the creator.
- You are responsible for making sure the software is updated when new releases, bugs are fixed and security vulnerabilities patched.
    - If the software has a release or security list, be on it.
    - Update your software in a timely manner.



Bonus: Ubuntu on Servers

CC BY 2.0: https://www.flickr.com/photos/pleia2/24965707290/

Firewall

We mentioned ufw for your desktop. This can also be used on a server, but there are a lot of reasons you may want to run a firewall on a server specifically:

- Typically directly on the internet with a publicly accessible IP address, rather than on a local LAN like personal systems tend to be
- Applications run on servers often serve data on the internet, making them more vulnerable and attractive to attack
- All addresses on the internet undergo constant attack by bots searching for common vulnerabilities to exploit

Check out tools like fail2ban to also automatically ban bots based on specific, repeated behavior, such as logging in with invalid ssh credentials.

Restrict SSH access

In your /etc/ssh/sshd_config, the following are common:

Disallow login by the root user:

```
PermitRootLogin no
```

Disallow logins except by SSH key:

```
PasswordAuthentication no
```

You may also have a list of the only users specifically allowed to log in:

```
AllowUsers elizabeth r2d2
```

Monitoring

Do you know when your system last rebooted?

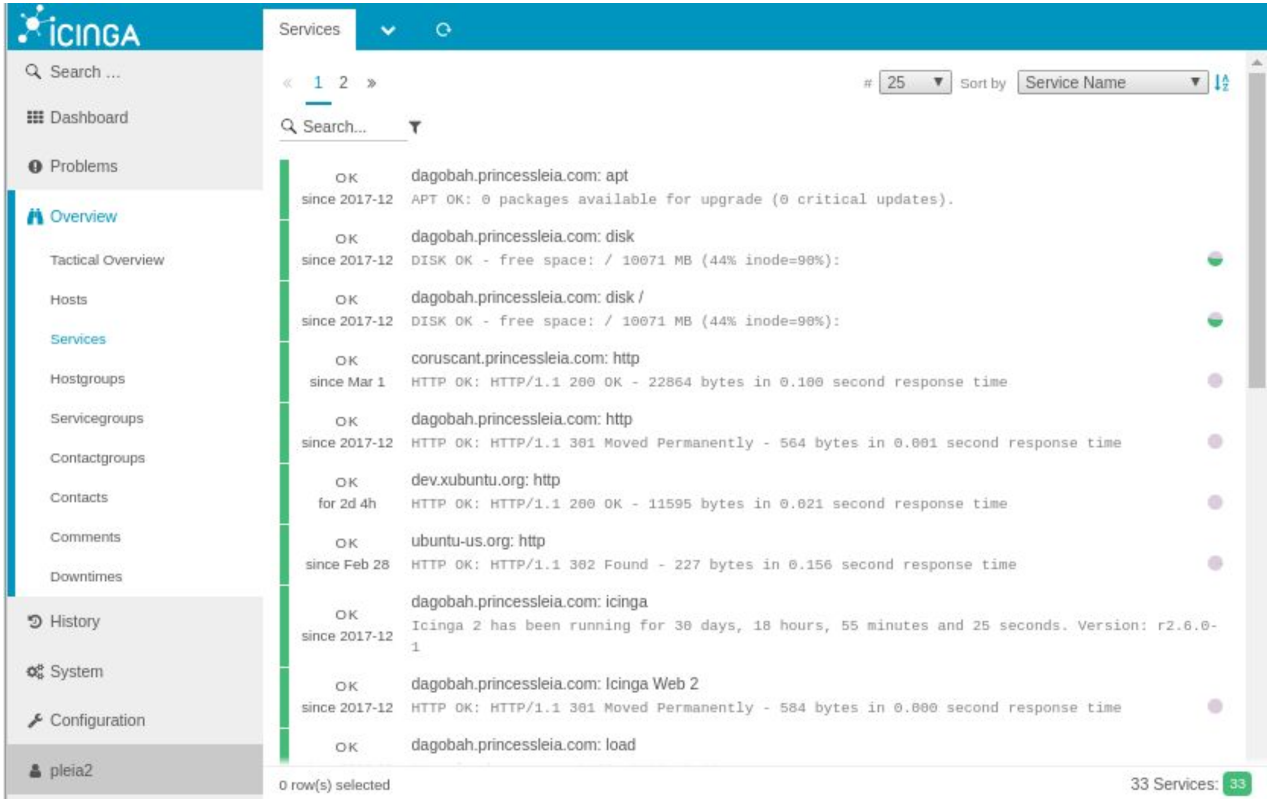
Did your MySQL database get killed for using too much Memory?

Is your website inaccessible because Apache didn’t reload after an upgrade?

Has your file system filled up without you noticing?

Use monitoring!

Monitoring example: Icinga2



25 Sort by Service Name

OK since 2017-12	dagobah.princessleia.com: apt	APT OK: 0 packages available for upgrade (0 critical updates).
OK since 2017-12	dagobah.princessleia.com: disk	DISK OK - free space: / 10071 MB (44% inode=90%):
OK since 2017-12	dagobah.princessleia.com: disk /	DISK OK - free space: / 10071 MB (44% inode=90%):
OK since Mar 1	coruscant.princessleia.com: http	HTTP OK: HTTP/1.1 200 OK - 22864 bytes in 0.100 second response time
OK since 2017-12	dagobah.princessleia.com: http	HTTP OK: HTTP/1.1 301 Moved Permanently - 564 bytes in 0.001 second response time
OK for 2d 4h	dev.xubuntu.org: http	HTTP OK: HTTP/1.1 200 OK - 11595 bytes in 0.021 second response time
OK since Feb 28	ubuntu-us.org: http	HTTP OK: HTTP/1.1 302 Found - 227 bytes in 0.156 second response time
OK since 2017-12	dagobah.princessleia.com: icinga	Icinga 2 has been running for 30 days, 18 hours, 55 minutes and 25 seconds. Version: r2.6.0-1
OK since 2017-12	dagobah.princessleia.com: Icinga Web 2	HTTP OK: HTTP/1.1 301 Moved Permanently - 584 bytes in 0.000 second response time
OK	dagobah.princessleia.com: load	

0 row(s) selected

33 Services: 33

Log-based alerts

Want to be informed when something unusual goes on? Use something like logcheck, install via the “logcheck” package in Ubuntu.

Logcheck sends an email when behavior from the logs is seen as a security problem or is unusual.

This tool requires tuning to ignore noise, but by using it you can gain familiarity with your system, the types of attacks that are common day to day and gives you log reading practice!

Update your web applications

Unlike desktop applications, web applications have long lagged behind formal packaging for distributions.

You will often be forced to install, upgrade and maintain these yourself.

Make sure you do this, *the most common attack vector on servers is through insecure applications*.

Questions? Additions?

Drop me an email at lyz@ubuntu.com!

Find talks, books, articles, blog, & more at princessleia.com

Thanks to: Nathan Haines, Mike Joseph, Steve Kowalik

I'm doing an *"Advanced Continuous Delivery Strategies for Containerized Applications Using DC/OS"* talk at 9:30AM on Friday here at SCaLE in the Container and Virtualization track.