LibreOffice
The Document Foundation

ROME
CONFERENCE

Security and Libreoffice

Jaskaran Veer Singh (jvsg)

jvsg1303@gmail.com

ROME | 11 October 2017

What this Presentation about?

- Emphasis on all things security
- Survey of existing security mechanisms
- What we do, and what we can do.
- For devs,corporations and paranoid people
- Focus on LO Core

What this presentation is not about?

- Bringing security secrets out in the open
- Exposing critical security bugs
- A defcon talk

Why care for security?

- Threats are rising and evolving
- Major establishments are now using Libreoffice
- Italian Defense Ministry is Libreoffice User!
- People are caring more for security
- “Scribbles” a tool developed by CIA

Learn from past mistakes

- Look up CVE database
- https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=LibreOffice
- Not just Libreoffice, but it’s dependencies too!
- Seems like few critical vulnerabilities
- But a lot of them are not made public!

Most of the LO vulnerabilities revolve around...

- Overflows
- Dangling Pointers
- Denial of Service (Crash)

Threats we face

- Denial of Service
- Getting hold of your system through a vulnerability in Libreoffice
- Theft of credentials (?)
- Bypassing protection

What are we doing currently...

Code Analyzers

- Coverity (since Oct 2012)
- Clang plugins
- Asan, Ubsan
- Crashtests

Coverity

- Since Oct 2012
- Dangling pointers
- Buffer overflows
- Memory corruption
- Careless use of signed values
- Defect Density of LO is the lowest among all the coverity projects

Yes, Size Matters

- Size tells a lot of information
- You can view the size of a file you dont have permission to even read. (In linux)
- Could guess the number of pages/slide
- Could tell if my presentation is long and boring, or short and interesting, even if you can’t read it.
- Can we fix this?

Add Bogus Pages?

- Pages that increase the size of the file, but don’t show up when you open them in Libreoffice
- Get average page size
- Get the number of bogus pages to be added
- Voila!
- But do we have to?

What could be done about security issues in the future?

Some Philosophies

- “Attachments are meant to be opened and links are meant to be followed”
- “Given enough eyeballs, all bugs are shallow”

A wiki page for All things security

- Page for the security enthusiasts, paranoid people and corporations.
- Instructions to build LO without potentially vulnerable modules (for extra security)
- Security Guidelines

Sandboxing can reduce damage

- SELinux Sandbox
- AppArmor
- Flatpak
- Ubuntu Snap
- AppVM

Sandbox – Under the hood

- Cgroups
- Namespaces
- Dbus for communication
- Additional stuff

SELinux Sandbox

- Introducing the SELinux Sandbox
- Just a simple c application
- Processes arguments and ensures the app specified is executed within the sandbox_t domain
- Looks like a simple interface “`sandbox libreoffice -blah`”
- BUT!
- By default permissions are only granted for STDIN and STDOUT
- You can grant permissions by:
- “`sandbox -X -H SandboxHome/ -t sandbox_web_t libreoffice -blah`”
- And so on….

SELinux Sandbox

- Libreoffice wont have access to various things like….
- Copy and Paste outside the application!!
- SELinux restricts it from using X server
- So, it would run inside nested X session.

SELINUX USER
HOW DID THAT HAPPEN?!?
memegenerator.net

AppArmor

- Easier than SELinux
- Only Works for Linux >= 2.6
- Apparently! Someone created AppArmor Profiles! Back in 2016
- Not sure if those are maintained now
- Dont quite look like Distro agnostic

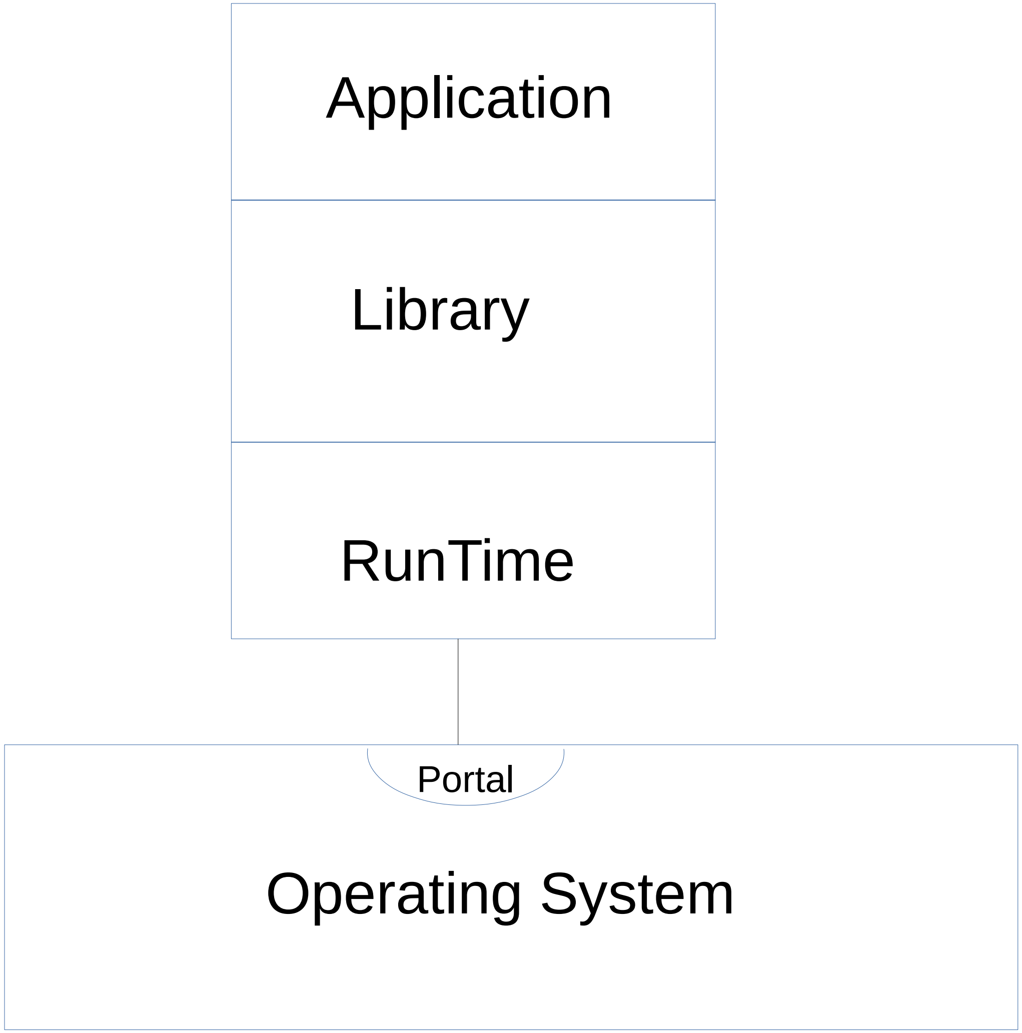
AppArmor

- Creating one is easy
- `Sudo apt-get install apparmor-utils`
- `sudo aa-genprof /path/to/libreoffice`
- This would log every apparmor event
- Then would ask you if you want to permit that event
- Would generate profile based on that

Flatpak

- One of the best and the easiest sandboxing techniques out there!
- Is only available for Linux
- Makes use of runtimes. Extensible too.
- Under the hood: A bublewrap facility.
- Doesnt include Java Runtime (JRE)
- Isn’t very stable

Flatpak Architecture



Ubuntu Snap

- Based on squashFS.
- Works for a lot of operating systems
- Read only File system, with a writable area.
- Fails Horribly for X11
- Works for Mir and Wayland (display servers)

The ultimate solution - VMs

- Spin up a VM and use Libreoffice inside it
- Could solve most of the issues
- Cumbersome
- Better alternative exists

Qubes OS

- You can run Libo on a virtual machine BUT…. you dont have to.
- Based on Xen Hypervisor and Linux
- The technology itself is called AppVM.
- Workspace is divided into Domains or “Doms”.
- Each Dom is made up of a “Template” and an application on top of it.
- Dom “Web browsers” can hold Chrome and Firefox and so on..
- Dom “Office stuff” can hold Libreoffice
- Multiple Libreoffice Vms for different types of files too.
- Can delete doms and create again if you think they are compromised

Qubes OS

- Multiple Libreoffice Vms for different types of files too.
- Can delete doms and create again if you think they are compromised
- Xen is tried and tested
- TBH, better in that regard than the new fads in the market everymonth.

Docker

- For fun experiment.
- GUI Apps can run in docker as well! Use VNC server (can be bundled in the docker image)
- Or do X11 forwarding
- Add these options when you do `docker run`
- `    -e DISPLAY=$DISPLAY`
- `    -v /tmp/.X11-unix:/tmp/.X11-unix`
- Hacks available to make it secure. But do it on your own risk.

Thanks for your time and attention!