USN-4903-1: curl vulnerability

Viktor Szakats discovered that curl did not strip off user credentials
from referrer header fields. A remote attacker could possibly use this
issue to obtain sensitive information.

How To: Disable Sleep/Hibernation on Ubuntu Server

In an earlier article, I wrote about making my own router. I am currently using Ubuntu’s server release for this. Imagine my surprise when I discovered a server configured to sleep!

Rather than try to figure out why a server would sleep by default, I’m just going to concentrate on fixing it. Seriously, though. Who decided to include all the sleep and hibernation underpinnings and decided they should be on by default? I can absolutely assure you that I did not turn them on!

*~*grumbles*~*

Seriously! My log contained lovely hints like:

[code]Apr 3 12:18:27 server systemd[1]: Reached target Sleep.[/code]

Disable Power Management – Ubuntu Server

This time you probably don’t need to open a terminal. It’s a server! Use SSH and you’re already in a terminal! Sheesh!

Anyhow, to make sure that this doesn’t happen again, let’s go ahead and kill everything that has to do with suspend, sleep, or hybrid-sleep. It’s actually pretty easy.

Let’s start with ‘sleep.target’ and we’re going to just mask these rather than removing them. If we mask them, we can unmask them – should we ever feel the need to do so, though I can’t see why you’d want them with server applications.

[code]sudo systemctl mask sleep.target[/code]

Next, let’s take care of ‘suspend.target’ with:

[code]sudo systemctl mask suspend.target[/code]

Then we’ll take care of ‘hibernate.target’ with:

[code]sudo systemctl mask hibernate.target[/code]

And, last but not least, we will go ahead and mask ‘hybrid-sleep.target’:

[code]sudo systemctl mask hybrid-sleep.target[/code]

As alluded to above, you can undo any of those by simply changing ‘mask’ to ‘unmask’ in the commands above and it will re-enable them. Why you’d want to do that, I have no idea – just like I have no idea why these things would even be included in a server-specific release, never mind why any of them would be enabled!

If you’re feeling so inclined, you can verify they’re off. For example, ‘sleep.target’ can be checked with:

[code]systemctl status sleep.target[/code]

Finally, thanks for reading. Like always, I love the feedback and the newsletter is still there waiting for you to sign up. If you do sign up, I chose a pretty crappy domain name and you should probably check your spam inbox for the confirmation email.

USN-4561-2: Rack vulnerabilities

USN-4561-1 fixed vulnerabilities in Rack. This update provides the
corresponding update for Ubuntu 16.04 LTS, Ubuntu 20.04 LTS and Ubuntu 20.10.
Original advisory details:

It was discovered that Rack incorrectly handled certain paths. An attacker
could possibly use this issue to obtain sensitive information. This issue
only affected Ubuntu 16.04 LTS, Ubuntu 18.04 LTS and Ubuntu 20.04 LTS.
(CVE-2020-8161)

It was discovered that Rack incorrectly validated cookies. An attacker
could possibly use this issue to forge a secure cookie. (CVE-2020-8184)

USN-4902-1: Django vulnerability

Dennis Brinkrolf discovered that Django incorrectly handled certain
filenames. A remote attacker could possibly use this issue to create or
overwrite files in unexpected directories.

USN-4901-1: Linux kernel (Trusty HWE) vulnerabilities

Adam Nichols discovered that heap overflows existed in the iSCSI subsystem
in the Linux kernel. A local attacker could use this to cause a denial of
service (system crash) or possibly execute arbitrary code. (CVE-2021-27365)
It was discovered that the LIO SCSI target implementation in the Linux
kernel performed insufficient identifier checking in certain XCOPY
requests. An attacker with access to at least one LUN in a multiple
backstore environment could use this to expose sensitive information or
modify data. (CVE-2020-28374)

Adam Nichols discovered that the iSCSI subsystem in the Linux kernel did
not properly restrict access to iSCSI transport handles. A local attacker
could use this to cause a denial of service or expose

sensitive information
(kernel pointer addresses). (CVE-2021-27363)

Adam Nichols discovered that an out-of-bounds read existed in the iSCSI
subsystem in the Linux kernel. A local attacker could use this to cause a
denial of service (system crash) or expose sensitive information (kernel
memory). (CVE-2021-27364)