The DirtyCOW bug that has been inherently residing in Linux source code of every kernel release since last eleven years has now been discovered to affect Android running on ARM chips too.

Before we go into the detail of how it affects Android app security, let us first understand how DirtyCOW attack works.

* Pick two files, one that can be written onto and one that cannot. Load them into memory with the kernel.
* Repeatedly write on the re-writable file and instruct the kernel every time to borrow memory from the other file that can’t be changed i.e. the read-only file.
* In a fraction of a second, the kernel will confuse memory buffer of the file you are writing into with that of the other file.
* This will result in the read-only file eventually being overwritten by you.

This bug can turn out to be disastrous for businesses, especially if the overwritten file is a configuration file or a system executable file. The reason why it can affect Android app security is because Android environment works on open-source Linux Operating System. The bug itself is a privilege escalation one, but remote attackers can combine it with other exploits and remotely attain root access on a computer.

A user on Github has come up with a proof-of-concept for a project to replace Android program known as “run-as”. “Run-as” works on the concept of Run-as on windows, and can allow an application to run as if it is done so by a different user. Though this is good for development and testing, it also gains root privilege by itself as soon as it starts and has the ability to pass the rootness to all apps it loads.

To avoid this, the “run-as” program by Google requires a phone to be rooted before it can be used to root. Replacing the “run-as” program that can only be managed by an admin with another version that can be managed by any user can clearly provide a means to permanently root a phone.

**Why Rooting can be Risky?**

Different people may have different motives behind rooting their phones. Some may do it for removing access programs installed with vendor software, some for applying security patches, and many others do it to for piracy purpose. Many a times this leads to careless installation of malicious applications that lead to security issues within a device. Hence, a rooted app can bypass sandboxing restriction on Android and get unauthorized access to log files, messages and other personally identifiable information.

To get your locked android phone genuinely rooted, DirtyCOW mechanism can be a better option till the time your vendor comes up with an Android app security update. System admins who manage a network of corporate systems may however, consider the risks of rooting too far than the benefits and would prefer not to be indulged in the game of rooted devices.