Perl 5.34.0 released

Version 5.34.0 of the Perl language has been released. "Perl 5.34.0 represents approximately 11 months of development since Perl 5.32.0 and contains approximately 280,000 lines of changes across 2,100 files from 78 authors." See this page for a list of changes; they include a new try/catch syntax, a new octal syntax, and many improvements to various modules.

Improving Application Security with UndefinedBehaviorSanitizer (UBSan) and GCC

The UBSan ("UndefinedBehaviorSanitizer") tool is a very useful, yet relatively unknown member of the GNU/Linux Toolchain family. This tool can improve the security of an application by efficiently detecting several types of errors in the source code. It is a run time tool that reports errors as the program executes.

In this article we introduce and discuss the features of UBSan; we explain how to use it; and we provide some tips and tricks showing how to get the maximum benefit from this tool.

What is Undefined Behavior and How Does it Impact Application Security?

What is undefined behavior? Language specifications often fail to say what the compiler should do when code does not conform to expected values. For example, the C specification does not say what the result should be if an array is indexed with an out-of-bounds value, or what should happen if the shift amount is greater than the size of the expression in a bitwise shift. Since the result for these cases is unspecified, the compiler is free to generate any code that produces the correct result when the values are within the correct range, and ignore the possibility of incorrect values. This means that different compilers may handle these situations differently and in general the result of undefined behavior is unpredictable. The code may "work" with one compiler on certain hardware and not with another combination, it may
"work" without optimization and fail with optimization or vice-versa. These situations generally point to an undefined behavior bug in the code.

- **QJSValue vs QJSManagedValue/QJSPrimitiveValue** [4]

  When Qt 6.1 got released you might have read about QJSManagedValue and how it gives more fine grained control over JavaScript execution?. But what does that actually mean? To understand this, let?s first recap what QJSValue is all about, and then compare it with the new classes.

- **The C Command Line Arguments Processing ? Linux Hint** [5]

  C programming language provides us with a way to pass command line arguments in Linux. This article will enlighten you more on this concept by discussing how the command line arguments are processed in C. Moreover, we will look at a suitable example that will possibly remove all your ambiguities regarding the said concept.

- **Calloc in C ? Linux Hint** [6]

  calloc? is yet another commonly used function of the ?stdlib.h? library. It stands for contiguous allocation. As the name says, this function is also used to allocate memory while using the heap instead of the stack. Again, this kind of memory allocation is known as dynamic memory allocation. The main aim of today?s discussion is to throw light on how the calloc? function works in C. Then, we will draw a comparison between the calloc? and malloc? functions. Finally, we will elaborate on the usage of the calloc? function in C by sharing an example with you.

- **How Do I See Directory Changes in Python? ? Linux Hint** [7]

  In some instances, especially in critical and restricted locations of the filesystem, it can be helpful to know when things change and what has changed. Using Linux Inotify tools and Python, we can view and log the changes that happen within the system.

  This tutorial will go over how to implement a simple script that uses Python and Linux Inotify API to monitor changes in a specific directory and log the console changes.
Before we get to the script, let us briefly discuss how Inotify works.

**Source URL:** [http://www.tuxmachines.org/node/151462](http://www.tuxmachines.org/node/151462)

**Links:**
[2] [https://lwn.net/Articles/856926/rss](https://lwn.net/Articles/856926/rss)
[4] [https://www.qt.io/blog/qjsvalue-vs-qjsmanagedvalue/qjsprimitivevalue](https://www.qt.io/blog/qjsvalue-vs-qjsmanagedvalue/qjsprimitivevalue)
[7] [https://linuxhint.com/see-directory-changes-python/](https://linuxhint.com/see-directory-changes-python/)