
08:15 - 08:45

DVA333

AUTOMATIC CONTROL OF SWITCHPORT STATUS/AUTOMATISK KONTROLL AV STATUS FÖR SWITCH-PORTAR

Dan Östergren

Advisor: Sara Lundahl

Examiner: Johan Åkerberg

Abstract:

"In large network environments today, it can be difficult to get a complete summary of how many switchports are in use. The solution that is usually used is manual login to the current switch for control, a step with time consumption and which is often performed on occasion. Information therefore risks being discovered too late with a measure that becomes reactive. Methods for making the information available in such a way that it can largely become proactive can instead enable timely action but also provide up-to-date information more easily accessible and ultimately save both time and expenses for a company.

The work focuses on the development of a solution with specific requirements with a focus on simplifying current steps; to be able to easily access up-to-date information on the status of switchports, to be able to get an overall picture of the distribution of switchports for a facility and the possibility of notification in the event of border crossings. Initially, an investigation is made of current solutions that are available in the area today, where several turn out to have shortcomings but also lack the necessary functions. As the solutions available today do not comply with the requirements, there is a motivation for developing a solution with its own functions. The solution that is put together is developed with separate functions for collecting information, interpreting, presenting and notifying information to the administrator. To ensure a stable solution with continuous operation, a server was installed in an external datacenter where a Linux distribution was used. Two different protocols for obtaining information are compared with practical tests, login via SSH, and SNMP-poll, where the chosen solution is based on SSH as a method due to previous experience from the author, but where SNMP shows more compatibility between manufacturers and models of hardware. Furthermore, the procedures of the functions that have been developed for the interpretation of collected information and the difficulties that have arisen in connection with this, for the avoidance of misinterpretations are described. Different options for presenting information to the administrator are compared, where the chosen solution was accessed via a website, this is due to the basic support regardless of platform. Something that, for example, an application cannot provide to the same extent. Among the various notification methods examined, the choice also fell in this case on a platform-independent method, where notification via e-mail was considered both easy to implement and with broad support among clients. Warnings for the specified criteria of the number of available switchports could thus be received by the administrator. All functions worked during the execution as planned and the solution is used by the customer. There are some suggestions for

improvement measures where SNMP can instead be used to advantage with broader support from other manufacturers and models, as well as protection functions when interpreting information.

"

Sammanfattning:

I stora nätverksmiljöer kan det idag vara svårt att få en komplett sammanställning av hur många switch-portar som är i bruk. Den lösning som vanligtvis används är manuell inloggning till aktuell switch för kontroll, ett moment med tidsåtgång och som ofta utförs vid enstaka tillfällen. Information riskerar därför att upptäckas för sent med en åtgärd som blir reaktiv. Metoder för att tillgängliggöra informationen på ett sådant sätt att den till stor del kan bli proaktiv kan istället möjliggöra åtgärder i tid men även tillhandahålla aktuell information mer lättillgängligt och i slutändan spara både tid och utgifter för ett företag.

08:50 - 09:35

DVA503

Evaluating the real-time capabilities of Zephyr on UP Squared - a pilot study with Zephyr and ACRN

Charlie Kotro; Jimmy Mäkeli

Advisor: Anna Friebe and Shaik Mohammed Salman

Examiner: Thomas Nolte

Abstract:

: In this study we have investigated if using Zephyr (User VM) and ACRN (Hypervisor) on the UP Squared (N3350) board are suitable for use in real-time scenarios. This was achieved by conducting a series of tests on Zephyr's scheduling API functions such as busy-wait and sleep under different system configurations. The tests involved running a task in isolation and taking timestamps of the current system cycle count before and after each function. To do this, we used the Common Trace Format functionality in Zephyr to collect execution traces, which were then analyzed using Babeltrace to convert the binary trace into human-readable form. The difference between the before and after timestamp were compared to the configured clock frequency to see if the system behaved as expected. We also evaluated how helpful the documentation was during the installation of the systems. Our findings in this report suggest that Zephyr and ACRN on the UP Squared board are not suitable for use in real-time scenarios due to large differences in system cycle count depending on the configuration.

Opponent: Magnus Sörensen

09:40 - 10:25

DVA428

Towards time predictable and efficient cache management in multi-threaded systems

Vildan Zivojevic

Advisor: Jakob Danielsson

Examiner: Moris Behnam

Abstract:

Once the cache memory was introduced in computer systems, the well-known gap in speeds between the memory and the CPU was reduced. However, various issues can occur within the cache, which has a significant impact on the performance and timing-predictability of an application. This thesis investigates one such issue, which is a cache contention. Most commonly, this problem can be detected inside of multi-core architecture, but also can be present within all systems that use a scheduler with multiple threads. In this thesis, we show a scenario where the cache contention occurs locally in the L1 data cache on a single-core, multi-threaded system. In this way, we will be able to examine the impact of local cache contention on system performance and timing-predictability. We furthermore mitigate cache contention through a way-based partitioning technique, where we propose a way to avoid cache contention, while still maintaining reasonable overall performance. Our results show that way-partitioning offers inter-thread isolation whilst showing a slight performance drop.

Opponent: Jimmy Mäkkeli