


<p>نام درس: سیستم‌های عامل پیشرفته نیم‌سال اول 87-88 استاد درس: دکتر رسول جلیلی تاریخ: 87/09/22 صفحه: 1</p>	<p>باسمه تعالی پروژه برنامه نویسی موعد تحویل: 87/10/30 (ساعت 23:59)</p>	 <p>دانشگاه صنعتی شریف دانشکده مهندسی کامپیوتر</p>
---	--	--

نکات قابل توجه:

- پاسخ‌های خود را به صورت الکترونیکی به آدرس Mahrooghi@ce.sharif.edu با عنوان [OS2:Project] ارسال نمایید.
- انجام این تمرین باید به صورت انفرادی انجام پذیرد.
- در صورت تاخیر در ارسال پاسخ، به ازای هر روز 20٪ نمره کل، کسر می‌گردد.
- تاریخ تحویل حضوری پروژه بعداً اعلام خواهد شد.

Goal :

Implement sharing mechanism among UML(User mode Linux) guest OS'es and/or a user process on the host kernel.

You will Implement a virtualized camera device front-end/back-end driver. The purpose of this project is to give you some kernel hacking experience and familiarize you with certain virtualization and sandboxing techniques.

Step 1.

Install and get familiar with User Mode Linux. Download User Mode Linux and get used to it.

<http://user-mode-linux.sourceforge.net/>


<http://ploug.eu.org/doc/l-uml-a4.pdf>

Step 2.

Familiarize yourselves with how normal non-virtualized drivers are written in Linux. Next, you will implement a simple version of a **virtualized** driver. Implement a front-end device driver in the Guest OS kernels. This front-end device driver will act as a camera device driver in the given Guest UML kernels. You should implement these functions. (We have no write() functionality here) open() / close() / read() / ioctl()

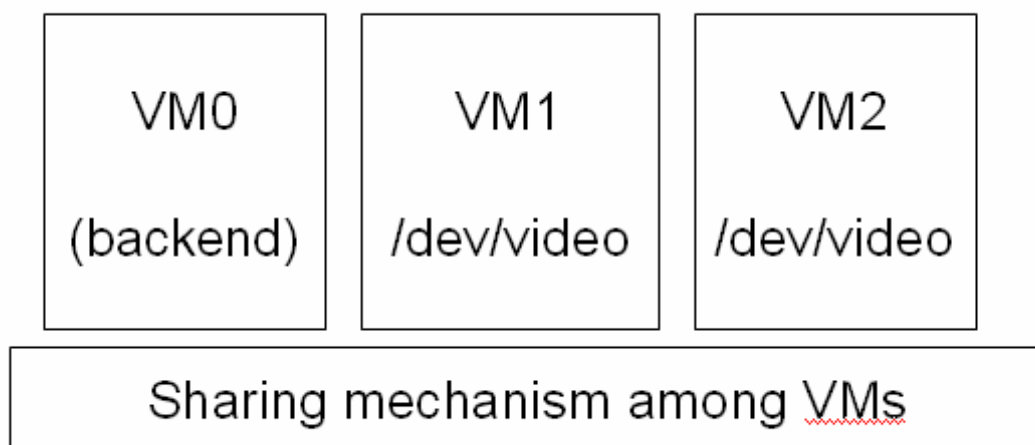
Step 3.

Implement a sharing mechanism among Guest UML kernel and the back end driver which will convey images from the backend driver from UML0 (or a user process if you prefer) to front-end drivers. This is the Key part of the project. You would need to research and Implement a mechanism to share data across multiple guest UML kernels and possibly a user process running on the host kernel. For the scenario where there are more than one UML instance running we recommend that implement separate queues in each UML instance and have the back-end poll these queues.

نام درس: سیستم‌های عامل پیشرفته نیم‌سال اول 87-88 استاد درس: دکتر رسول جلیلی تاریخ: 87/09/22 صفحه: 2	باسمه تعالی پروژه برنامه نویسی موعده تحویل: 87/10/30 (ساعت 23:59)	 دانشگاه صنعتی شریف دانشکده مهندسی کامپیوتر
--	--	--

Step 4.

Implement backend device driver in VM0 or a user process running on the host kernel which sends its image data to the other front-end drivers. Backend driver sends the image data to the front-end driver using the sharing mechanism. The backend driver should respond to the open/close/read/ioctl calls of the front end. You may “fake” the camera device with a sample Bitmap image. Additionally you would need to implement a Scaling function in the back end driver which provides a scaled down version of the image extracted from your “fake” camera’s buffer.



This project was inspired from an assignment given in CS6210 by Dr. Karsten Schwan, Computing College, Georgia Tech.

آنچه باید تحویل داده شود :

- کدهای نوشته شده همراه با مستندات کامل
- گزارش کامل نحوه انجام مراحل پروژه