现如今，Linux操作系统因为其完备的功能，强大的性能、优秀的兼容性以及开源免费等优点，已被人们广泛应用在高性能服务器甚至是个人电脑上，但是，如同Windows操作系统一样，Linux的广泛应用必然会吸引大量的、有针对性的恶意攻击行为，所以Linux在提供强大功能的同时，如何保证较高的安全性也成为Linux开发者和研究人员的一大难题。

在Linux操作系统中，最为核心的部分是内核。内核的安全与否，对整个计算机系统的安全与稳定来说格外的重要。但是近些年，在Linux功能越来越完善的同时，其内核漏洞也层出不穷，有些内核函数的设计缺陷成为了攻击者利用来入侵系统的工具，另外，Linux为了实现对旧功能和各类设备的兼容，在内核中保留了大量过时的函数，这些老旧的代码也成为了威胁整个系统的安全隐患。

更进一步说，对于系统中运行的某一个特定的进程，内核中的绝大多数的函数都不会被它使用到，但是许多攻击者都喜欢利用那些本身具有特权的服务进程（如ftp、ssh和http等等），使用控制流劫持等攻击手段，改变这些进程的正常运行轨迹，然后利用内核函数来完成恶意行为。自然地，通过限定进程访问内核函数的范围，成为了解决这些问题的一个可行思路。

所以我们希望能够用一定的方法，通过限定具有特权的服务进程所能访问的内核函数范围，来防御控制流劫持等利用内核函数漏洞进行的恶意攻击，来有效的提高Linux内核安全性。

现存的一些较为成熟的Linux系统防护方法主要有：内核最小化，基于主机的异常检测系统，访问控制技术，虚拟化与沙箱技术等等，它们各有各的优缺点。

内核裁剪与最小化技术主要是依据用户需要的功能，去掉冗余代码，定制出最小化的内核，大多应用在嵌入式系统中。这种方法能够删除系统中不必要的功能，从而很大程度的提高内核的安全性，但是这一般意味着需要重新编译内核，在实际应用中并不方便。

异常检测系统则使用事先定义的正常的行为来判断进程当下行为的正常与否，可融合许多高效的人工智能技术，并且可以检测出新的攻击类型。但是它们大多大多基于用户空间设计，并没有深入内核，同时长期被较高的误报率所困扰，也存在许多方法能绕过其检测。

访问控制技术使用一定的安全策略，对资源等的访问加以控制，防止无权限或低权限用户访问关键、敏感的数据，具有性能开销小，效率高等特点，但是合理且正确的安全策略的制定难度则相对较大。

虚拟化技术在宿主机器上虚拟化出全部或部分的运行环境，来隔离系统和程序的运行。沙箱技术则是部分结合访问控制与虚拟化的思想，根据预设安全策略，构建一个可控的运行环境，将程序所有或部分行为限制在沙箱内，使其不会对真实系统造成任何不良影响。它们的优缺点类似：它们都可以完全或部分隔离系统或单个可疑程序的运行，安全性高。但是构建出可控环境会带来较大的系统性能开销，对沙箱来说，如何制定合适的安全策略也是一大难点。

我们根据上述方法的优缺点，结合异常检测系统、访问控制和沙箱的部分思想，提出基于内核函数异常调用监测的系统防护思想，总结为一句话：实时监控特权服务进程不应该调用的部分内核函数，来防止攻击者利用这些进程，组织内核函数进行攻击，对内核造成威胁。我们将会设计并实现该系统，并最后部署在我们自己课题组的服务器上，提高实际服务器系统的安全性。该选题同时具有理论和实践的意义。

参考文献：

1. Stanley, D.M. Improved kernel security through code validation, diversification, and minimization [D]. Purdue University, 2013.
2. Anil Kurmus, Alessandro Sorniotti, Rüdiger Kapitza. Attack surface reduction for commodity OS kernels：Trimmed garden plants may attract less bugs [A]. The European Workshop on Systems Security (EuroSec), 2011.
3. Shayan Eskandari, Wael Khreich, Syed Shariyar Murtaza, Abdelwahab Hamou-Lhadj and Mario Couture. Monitoring system calls for anomaly detection in modern operating systems [A]. IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), 2013.
4. Anil Kurmus, Reinhard Tartler, Daniela Dorneanu, Bernhard Heinloth, Valentin Rothberg, Andreas Ruprecht, Wolfgang Schr¨oder-Preikschat, Daniel Lohmann, and R¨udiger Kapitza. Attack surface metrics and automated compile-time OS kernel tailoring [J]. IBM Corporation, 2013.
5. Jia-Ju Bai, Hu-Qiu Liu, Yu-Ping Wang, Shi-Min Hu. Complete runtime tracing for device drivers based on LLVM [A]. IEEE 39th Annual International Computers, Software & Applications Conference, 2015.
6. Rafik Fahem and Michel Dagenais. Efficient conditional tracepoints in kernel space [J]. The Open Cybernetics & Systemics Journal, 2012.
7. Steven A. Hofmeyr, Stephanie Forrest, Anil Somayaji. Intrusion detection using sequences of system calls [J]. Journal of Computer Security, 1999.
8. Syed Shariyar Murtaza, Wael Khreich, Abdelwahab Hamou-Lhadj, Mario Couture. A Host-based anomaly detection approach by representing system calls as states of kernel modules. IEEE International Symposium on Software Reliability Engineering, 2013.
9. Waqas Haider, Jiankun Hu, Xinghuo Yu, Yi Xie. Integer data zero-watermark assisted system calls abstraction and normalization for host based anomaly detection systems [A]. IEEE 2nd International Conference on Cyber Security and Cloud Computing, 2015.
10. Goverdhan Reddy Jidiga, Dr.P.Sammulal. Anomaly detection using smart tracing tricks on call stack [A]. International Conference for Convergence of Technology, 2014.
11. Kui Xu, Danfeng (Daphne) Yao, Barbara G. Ryder, Ke Tian. Probabilistic program modeling for high-precision anomaly classification. IEEE 28th Computer Security Foundations Symposium, 2015.
12. Jonathon Ng, Deepti Joshi, Shankar M. Banik. Applying data mining techniques to intrusion detection [A]. 12th International Conference on Information Technology, 2015.
13. Gideon Creech, Jiankun Hu. A semantic approach to host-based intrusion detection systems using contiguous and discontiguous system call patterns [J]. IEEE Transactions on Computers, 2014.
14. https://www.kernel.org/doc/Documentation/trace/ftrace.txt
15. https://www.kernel.org/doc/Documentation/kprobes.txt
16. http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-5195
17. Muhammad Shams Ul haq, Lejian Liao, Ma Lerong, Design and implementation of sandbox technique for isolated applications [A]. IEEE Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2016.
18. Tomohiro Shioya, Yoshihiro Oyama, and Hideya Iwasaki. A sandbox with a dynamic policy based on execution contexts of applications [A]. Advances in Computer Science-asian Computer & Network Security, Asian Computing Science Conference, 2007.
19. Ma Bo, Mu Dejun, Fan Wei, Hu Wei. Improvements the Seccomp sandbox based on PBE theory [A]. 27th International Conference on Advanced Information Networking and Applications Workshops, 2013.
20. Ján Hurtuk, Anton Baláž, Norbert Ádám. Security sandbox based on RBAC model [A]. 11th IEEE International Symposium on Applied Computational Intelligence and Informatics, 2016.
21. Yoshihiro Oyama, Koichi Onoue. Speculative security checks in sandboxing systems [A]. 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS’05), 2005.
22. Mohan Rajagopalan, Matti A. Hiltunen, Trevor Jim, and Richard D. Schlichting. System call monitoring using authenticated system calls [J]. IEEE Transactions on Dependable & Secure Computing, 2006.
23. Muhammad Shams Ul haq, Lejian Liao, Ma Lerong. Transitioning native application into virtual machine by using hardware virtualization extensions [A]. International Symposium on Computer, Consumer and Control, 2016.
24. Z. Cliffe Schreuders, Christian Payne. Reusability of functionality-based application confinement policy abstractions. International Conference on Information & Communications Security, 2008.
25. 袁源，戴冠中.LKM后门综述 [J]. 计算机科学, 2008.
26. 蒋卫华，李伟华，杜君.缓冲区溢出攻击：原理，防御及检测 [J]. 计算机工程, 2003.
27. 黄志军，郑涛.基于Return-Oriented Programming的程序攻击与防护 [J]. 计算机科学, 2012.
28. 黄义文.Linux操作系统内核裁剪的分析 [J]. 中国民航飞行学院学报, 2010.
29. 徐晨辉.嵌入式Linux内核裁剪及移植的研究与实现 [D]. 上海：东华大学, 2009.
30. 田俊峰，张喆，赵卫东.基于误用和异常技术相结合的入侵检测系统的设计与研究 [J]. 电子与信息学报, 2006.
31. 黄飞.基于进程行为的主机异常检测系统 [D]. 扬州：扬州大学, 2008.
32. 李晓琦，刘奇旭，张玉清.基于模拟攻击的内核提权漏洞自动利用系统 [J]. 中国科学院大学学报, 2015.
33. 冯培钧，张平，陈志峰，张擂.一种新型Linux内核级Rootkit设计与实现 [J]. 信息工程大学学报, 2016.
34. 赵旭，陈丹敏，颜学雄，王清贤.沙箱技术研究综述 [J]. 中原工学院学报, 2014.
35. 程香鹏，陈莉君.基于LSM的沙箱模块设计与实现 [J]. 计算机与数字工程, 2014.
36. 白彦芳，魏立峰，丁传新.基于完整性的执行控制设计与实现. 2012.
37. 程龙，杨小虎.Linux系统内核的沙箱模块实现 [J]. 计算机应用, 2004.