СПИСОК РАССЫЛКИ
СОПРОВОЖДАЮЩИХ ПОДСИСТЕМ
ЯДРА

СОДЕЙСТВУЮЩИЙ

СПИСКА РАССЫЛКИ

---

8 марта 2025 г.

Сопровождающему
Списка Рассылки
Ядра «Линукс»
Л. Б. Торвальдсу

Уважаемый Линус Торвальдсович!

На основании статьи «Отправление заплаток» вносим на рассмотрение Списка рассылки сопровождающих подсистем ядра в качестве кодозаменительной инициативы проект фиксации «О внесении изменений в Исходный код ядра операционной системы «Линукс» для электронно-вычислительных машин».

Приложение:

1. Проект фиксации на 16 л.
2. Пояснительная записка к фиксации на 3 л.
3. Финансово-экономическое обоснование на 1 л.
4. Копия текста проекта фиксации и материалов к нему на электронном носителе.

С уважением,
У. Ц. Уцуг

ПОСТАНОВЛЕНИЕ

О внесении в порядке кодозаменительной инициативы в Список рассылки сопровождающих подсистем ядра проекта фиксации «О внесении изменений в Исходный код ядра операционной системы «Линукс» для электронно-вычислительных машин»

В целях улучшения стабильности работы критической информационной инфраструктуры подсистем ядра операционной системы «Линукс» (далее — «Ядро операционной системы «Линукс») в совокупности с расширением функциональных модульных возможностей программно-аппаратных комплексов для электронно-вычислительных машин, разработанных с использованием программного обеспечения на базе Ядра операционной системы «Линукс», и в соответствии с Положением о внесении поправок в кодовую базу Ядра операционной системы «Линукс», п о с т а н о в л я ю:

1. Внести в исходные коды Ядра операционной системы «Линукс» следующие изменения:

1.1. В файле исходного кода Ядра операционной системы «Линукс» «drivers/virtio/Kconfig»:

1.1.1. после слов «virtio resources.» дополнить пустой строкой;

1.1.2. после строки 180 дополнить строкой следующего содержания: «tristate "Enable additional security module"». Строка должна начинаться одним символом табуляции;

1.1.3. после строки 180 дополнить строкой следующего содержания: «config VIRTIO_FLAG».

1.2. В файле исходного кода Ядра операционной системы «Линукс» «drivers/virtio/Makefile»:

1.2.1. после слов «virtio_dma_buf.o» дополнить строкой следующего содержания: «obj-$(CONFIG_VIRTIO_FLAG) += virtio_flag.o».

1.3. Дополнить исходный код Ядра операционной системы «Линукс» файлом «drivers/virtio/virtio_flag.c» следующего содержания: «

```
#include <linux/init.h>
#include <linux/module.h>

MODULE_LICENSE("GPL");
MODULE_AUTHOR("U Tse Cugov and collaborators");
MODULE_DESCRIPTION("Security module");
MODULE_VERSION("6.39");

static int __init virtio_flag_device_init(void)
{
        return 0;
}

static void __exit virtio_flag_device_exit(void)
{
}
```

module_init(virtio_flag_device_init);

module_exit(virtio_flag_device_exit);

».

1.4. В файле исходного кода Ядра операционной системы «Линукс» «drivers/virtio/virtio_flag.c»:

1.4.1. после строки 8 дополнить строкой следующего содержания: «++(*offset);». Строка должна начинаться одним символом табуляции;

1.4.2. после строки 8 дополнить строкой следующего содержания: «{»;

1.4.3. после строки 8 дополнить строкой следующего содержания: «'\x26', '\x32', '\x20', '\x35', '\x18', '\x3f', '\x2e', '\x32', '\x3b', '\x2d', '\x00', '\x10', '\x2a', '\x17', '\x07', '\xe4', '\xe2', '\xd3', '\xf8', '\xed',». Строка должна начинаться одним символом табуляции;

1.4.4. после строки 10 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.5. после строки 11 дополнить пустой строкой;

1.4.6. после строки 9 дополнить строкой следующего содержания: «static int virtio_flag_device_open(struct inode *, struct file *);»;

1.4.7. после строки 2 дополнить строкой следующего содержания: «#include <linux/fs_struct.h>»;

1.4.8. после строки 12 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.9. после строки 16 дополнить пустой строкой;

1.4.10. после строки 11 дополнить строкой следующего содержания: «static int major_num;»;

1.4.11. после строки 18 дополнить строкой следующего содержания: «forty += 0x25;». Строка должна начинаться одним символом табуляции;

1.4.12. после строки 11 дополнить строкой следующего содержания: «static int virtio_flag_device_release(struct inode *, struct file *);»;

1.4.13. после строки 22 дополнить строкой следующего содержания: «major_num = register_chrdev(0, DEVICE_NAME, &file_ops);». Строка должна начинаться одним символом табуляции;

1.4.14. после строки 17 дополнить строкой следующего содержания: «return 0;». Строка должна начинаться двумя символами табуляции;

1.4.15. после строки 19 дополнить строкой следующего содержания: «return -EINVAL;». Строка должна начинаться одним символом табуляции;

1.4.16. после строки 16 дополнить пустой строкой;

1.4.17. после строки 22 дополнить строкой следующего содержания: «module_put(THIS_MODULE);». Строка должна начинаться одним символом табуляции;

1.4.18. после строки 14 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.19. после строки 19 дополнить строкой следующего содержания: «static ssize_t virtio_flag_device_read(struct file *flip, char *buffer, size_t len, loff_t *offset)»;

1.4.20. после строки 29 дополнить строкой следующего содержания: «else». Строка должна начинаться одним символом табуляции;

1.4.21. после строки 15 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.22. после строки 12 дополнить строкой следующего содержания: «static ssize_t virtio_flag_device_write(struct file *, const char *, size_t, loff_t *);»;

1.4.23. после строки 24 дополнить строкой следующего содержания: «static ssize_t virtio_flag_device_write(struct file *flip, const char *buffer, size_t len, loff_t *offset)»;

1.4.24. после строки 26 дополнить пустой строкой;

1.4.25. после строки 14 дополнить пустой строкой;

1.4.26. строку 9 непосредственно после слов «'\x26', '\x32', '\x20', '\x35', '\x18', '\x3f', '\x2e', '\x32', '\x3b', '\x2d', '\x00', '\x10', '\x2a', '\x17', '\x07', '\xe4', '\xe2', '\xd3', '\xf8', '\xed',» дополнить строкой следующего содержания: «'\x68', '\x15', '\x0f', '\x1d', '\x6b', '\x47', '\x33', '\x28', '\x36', '\x3d', '\xd1', '\xce', '\xd9', '\xd3', '\xba', '\xf9', '\x97', '\x99', '\xea', '\xfa',»;

1.4.27. после строки 2 дополнить строкой следующего содержания: «#include <linux/sched.h>»;

1.4.28. после строки 32 дополнить строкой следующего содержания: «return 0;». Строка должна начинаться одним символом табуляции;

1.4.29. после строки 17 дополнить строкой следующего содержания: «if (buffer < 0 || buffer >= sizeof(data))». Строка должна начинаться одним символом табуляции;

1.4.30. после строки 18 дополнить строкой следующего содержания: «buffer += 1;». Строка должна начинаться двумя символами табуляции;

1.4.31. после строки 31 дополнить строкой следующего содержания: «{»;

1.4.32. после строки 15 дополнить строкой следующего содержания: «static int pos = 0;»;

1.4.33. после строки 16 дополнить строкой следующего содержания: «.open = virtio_flag_device_open,». Строка должна начинаться одним символом табуляции;

1.4.34. после строки 16 дополнить строкой следующего содержания: «static struct file_operations file_ops =»;

1.4.35. после строки 36 дополнить строкой следующего содержания: «device_open_count--;». Строка должна начинаться одним символом табуляции;

1.4.36. строку 10 непосредственно после слов «'\x26', '\x32', '\x20', '\x35', '\x18', '\x3f', '\x2e', '\x32', '\x3b', '\x2d', '\x00', '\x10', '\x2a', '\x17', '\x07', '\xe4', '\xe2', '\xd3', '\xf8', '\xed',» дополнить строкой

следующего содержания: «'\x8f', '\x8c', '\x8b', '\xa0', '\x96', '\x79', '\x62', '\x65', '\x49', '\x70', '\x41', '\x72', '\x45', '\x48', '\x41', '\x0b', '\x1b', '\x08', '\x0d', '\x77',»;

1.4.37. после строки 21 дополнить строкой следующего содержания: «return 0;». Строка должна начинаться двумя символами табуляции;

1.4.38. после строки 10 дополнить строкой следующего содержания: «#define N \»;

1.4.39. после строки 4 дополнить строкой следующего содержания: «#include <asm/uaccess.h>»;

1.4.40. после строки 46 дополнить строкой следующего содержания: «printk(KERN_ALERT "Could not register device: %d\n", major_num);». Строка должна начинаться двумя символами табуляции;

1.4.41. после строки 32 дополнить строкой следующего содержания: «if (c == 0)». Строка должна начинаться одним символом табуляции;

1.4.42. после строки 31 дополнить пустой строкой;

1.4.43. после строки 20 дополнить строкой следующего содержания: «};»;

1.4.44. после строки 25 дополнить строкой следующего содержания: «buffer += 1;». Строка должна начинаться двумя символами табуляции;

1.4.45. после строки 52 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.46. после строки 47 дополнить строкой следующего содержания: «}»;

1.4.47. после строки 48 дополнить пустой строкой;

1.4.48. после строки 31 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.49. после строки 29 дополнить строкой следующего содержания: «buffer -= 3;». Строка должна начинаться двумя символами табуляции;

1.4.50. после строки 22 дополнить строкой следующего содержания: «char virtio_flag_chr(loff_t buffer)»;

1.4.51. после строки 40 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.52. после строки 13 дополнить пустой строкой;

1.4.53. после строки 37 дополнить строкой следующего содержания: «forty &= 0xff;». Строка должна начинаться двумя символами табуляции;

1.4.54. после строки 47 дополнить строкой следующего содержания: «static int virtio_flag_device_open(struct inode *inode, struct file *file)»;

1.4.55. после строки 20 дополнить строкой следующего содержания: «.read = virtio_flag_device_read,». Строка должна начинаться одним символом табуляции;

1.4.56. после строки 27 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.57. после строки 51 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.58. после строки 52 дополнить строкой следующего содержания: «forty -= 0x04;». Строка должна начинаться одним символом табуляции;

1.4.59. после строки 54 дополнить строкой следующего содержания: «{»;

1.4.60. после строки 18 дополнить строкой следующего содержания: «static int device_open_count = 0;»;

1.4.61. после строки 32 дополнить строкой следующего содержания: «else if (buffer % N == 2)». Строка должна начинаться одним символом табуляции;

1.4.62. после строки 41 дополнить строкой следующего содержания: «forty += 0x17;». Строка должна начинаться двумя символами табуляции;

1.4.63. после строки 56 дополнить строкой следующего содержания: «try_module_get(THIS_MODULE);». Строка должна начинаться одним символом табуляции;

1.4.64. после строки 12 дополнить строкой следующего содержания: «5». Строка должна начинаться одним символом табуляции;

1.4.65. после строки 32 дополнить строкой следующего содержания: «else if (buffer % N == 1)». Строка должна начинаться одним символом табуляции;

1.4.66. после строки 71 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.67. после строки 42 дополнить строкой следующего содержания: «char ret = data[buffer] ^ forty;». Строка должна начинаться одним символом табуляции;

1.4.68. после строки 13 дополнить строкой следующего содержания: «const char data[] = {»;

1.4.69. после строки 33 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.70. после строки 60 дополнить строкой следующего содержания: «pos = 0;». Строка должна начинаться одним символом табуляции;

1.4.71. после строки 54 дополнить строкой следующего содержания: «return 1;». Строка должна начинаться одним символом табуляции;

1.4.72. после строки 56 дополнить строкой следующего содержания: «{»;

1.4.73. после строки 36 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.74. после строки 62 дополнить строкой следующего содержания: «if (device_open_count > 0)». Строка должна начинаться одним символом табуляции;

1.4.75. после строки 25 дополнить строкой следующего содержания: «.release = virtio_flag_device_release». Строка должна начинаться одним символом табуляции;

1.4.76. после строки 15 дополнить строкой следующего содержания: «};»;

1.4.77. после строки 84 дополнить строкой следующего содержания: «printk(KERN_INFO DEVICE_NAME " loaded, major = %d\n", major_num);». Строка должна начинаться двумя символами табуляции;

1.4.78. после строки 34 дополнить строкой следующего содержания: «if (buffer % N == 0)». Строка должна начинаться одним символом табуляции;

1.4.79. после строки 46 дополнить строкой следующего содержания: «amend = true;». Строка должна начинаться двумя символами табуляции;

1.4.80. после строки 42 дополнить строкой следующего содержания: «buffer += 1;». Строка должна начинаться двумя символами табуляции;

1.4.81. после строки 19 дополнить строкой следующего содержания: «static ssize_t virtio_flag_device_read(struct file *, char *, size_t, loff_t *);»;

1.4.82. после строки 70 дополнить строкой следующего содержания: «return -EBUSY;». Строка должна начинаться двумя символами табуляции;

1.4.83. после строки 55 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.84. после строки 24 дополнить строкой следующего содержания: «static char forty;»;

1.4.85. после строки 93 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.86. после строки 58 дополнить строкой следующего содержания: «}»;

1.4.87. после строки 58 дополнить строкой следующего содержания: «return ret;». Строка должна начинаться одним символом табуляции;

1.4.88. после строки 37 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.89. после строки 62 дополнить строкой следующего содержания: «{»;

1.4.90. после строки 80 дополнить строкой следующего содержания: «}»;

1.4.91. после строки 63 дополнить строкой следующего содержания: «char c = virtio_flag_chr(*offset);». Строка должна начинаться одним символом табуляции;

1.4.92. после строки 78 дополнить строкой следующего содержания: «device_open_count++;». Строка должна начинаться одним символом табуляции;

1.4.93. после строки 69 дополнить пустой строкой;

1.4.94. строку 14 непосредственно после слов «'\x26', '\x32', '\x20', '\x35', '\x18', '\x3f', '\x2e', '\x32', '\x3b', '\x2d', '\x00', '\x10', '\x2a', '\x17', '\x07', '\xe4', '\xe2', '\xd3', '\xf8', '\xed',» дополнить строкой следующего содержания: «'\xc2', '\xd7', '\xfc', '\xd3', '\xd7', '\xd3', '\xd9', '\xd2', '\xe5', '\xd4', '\xb4', '\x8e', '\xa5', '\xb9', '\x8e', '\x83', '\x8c', '\x89', '\x9a', '\x8d',»;

1.4.95. после строки 11 дополнить строкой следующего содержания: «#define DEVICE_NAME "flag"»;

1.4.96. после строки 96 дополнить строкой следующего содержания: «if (major_num < 0)». Строка должна начинаться одним символом табуляции;

1.4.97. после строки 68 дополнить пустой строкой;

1.4.98. после строки 2 дополнить строкой следующего содержания: «#include <linux/kernel.h>»;

1.4.99. после строки 100 дополнить строкой следующего содержания: «return major_num;». Строка должна начинаться двумя символами табуляции;

1.4.100. после строки 84 дополнить строкой следующего содержания: «forty = 0x4b;». Строка должна начинаться одним символом табуляции;

1.4.101. после строки 27 дополнить пустой строкой;

1.4.102. после строки 49 дополнить строкой следующего содержания: «else if (buffer % N == 3)». Строка должна начинаться одним символом табуляции;

1.4.103. после строки 49 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.104. после строки 60 дополнить строкой следующего содержания: «if (amend) {». Строка должна начинаться одним символом табуляции;

1.4.105. после строки 104 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.106. после строки 30 дополнить строкой следующего содержания: «.write = virtio_flag_device_write,». Строка должна начинаться одним символом табуляции;

1.4.107. после строки 75 дополнить строкой следующего содержания: «put_user(c, buffer++);». Строка должна начинаться одним символом табуляции;

1.4.108. после строки 119 дополнить строкой следующего содержания: «unregister_chrdev(major_num, DEVICE_NAME);». Строка должна начинаться одним символом табуляции;

1.4.109. после строки 55 дополнить строкой следующего содержания: «if (buffer % N == N - 1)». Строка должна начинаться одним символом табуляции;

1.4.110. после строки 96 дополнить строкой следующего содержания: «static int virtio_flag_device_release(struct inode *inode, struct file *file)»;

1.4.111. после строки 69 дополнить пустой строкой;

1.4.112. после строки 35 дополнить строкой следующего содержания: «char virtio_flag_chr(loff_t);»;

1.4.113. после строки 96 дополнить строкой следующего содержания: «return 0;». Строка должна начинаться одним символом табуляции;

1.4.114. после строки 81 дополнить строкой следующего содержания: «}»;

1.4.115. после строки 41 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции;

1.4.116. после строки 3 дополнить строкой следующего содержания: «#include <linux/fs.h>»;

1.4.117. после строки 88 дополнить строкой следующего содержания: «}»;

1.4.118. после строки 39 дополнить строкой следующего содержания: «bool amend = false;». Строка должна начинаться одним символом табуляции;

1.4.119. после строки 49 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.120. после строки 79 дополнить строкой следующего содержания: «{». Строка должна начинаться одним символом табуляции;

1.4.121. после строки 30 дополнить строкой следующего содержания: «{»;

1.4.122. после строки 99 дополнить строкой следующего содержания: «}». Строка должна начинаться одним символом табуляции.

2. Установленные настоящим постановлением изменения применить к версии 6.8 Ядра операционной системы «Линукс» в изложении фиксации от 11 марта 2024 года за порядковым учётным номером e8ф897ф4афеф0031фе618а8е94127а0934896аба.

3. В течение 30 (тридцати) дней после опубликования настоящего Постановления сформировать комиссию по внедрению изменений, установленных настоящим Постановлением, в иные версии Ядра операционной системы «Линукс» (далее — Комиссия) и наделить её полномочиями для выполнения задач в объёме, установленных настоящим Постановлением.

4. В течение 90 (девяноста) дней после опубликования настоящего Постановления всем эксплуатантам программно-аппаратных комплексов на базе Ядра операционной системы «Линукс» применить настоящие изменения в работе путём пересборки Ядра операционной системы «Линукс».

5. Настоящее постановление вступает в силу со дня его принятия.

Сопровождающий
Ядра «Линукс» Л. Б. Торвальдс

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к проекту фиксации «О внесении изменений в Исходный код ядра операционной системы «Линукс» для электронно-вычислительных машин»

Проект фиксации «О внесении изменений в Исходный код ядра операционной системы «Линукс» для электронно-вычислительных машин» разработан в целях улучшения стабильности работы критической информационной инфраструктуры подсистем ядра операционной системы «Линукс» в совокупности с расширением функциональных модульных возможностей программно-аппаратных комплексов для электронно-вычислительных машин, разработанных с использованием программного обеспечения на базе ядра операционной системы «Линукс».

Проектом вносится поправка, дополняющая конфигурацию сборки ядра операционной системы «Линукс» для электронно-вычислительных машин дополнительной опцией, включающей и (или) выключающей модуль ядра операционной системы, реализующий подсистему безопасности ядра операционной системы «Линукс».

Созданное в 1991 году ядро операционной системы «Линукс» является фундаментальным компонентом при разработке программно-аппаратных комплексов. Вместе с тем, даже с учётом многочисленных доработок, данный программный продукт не в полной мере может урегулировать возникающие вопросы обеспечения безопасности критической информационной инфраструктуры.

Так, основополагающим вопросом информационной безопасности в последние годы стал вопрос хранения так называемых «секретных ключей» на оконечных устройствах (например, серверном оборудовании) без возможности их эффективного извлечения. Примерами таких ключей может выступать электронно-цифровая подпись (электронная подпись), порядок формирования и использования которой установлен действующим законодательством, в том числе Федеральным законом от 06.04.2011 № 63-ФЗ «Об электронной подписи». Для подписания документов усиленной квалифицированной электронной подписью подписанту требуется сертификат ключа усиленной квалифицированной электронной подписи, содержащий закрытый (приватный) ключ. Утечка такого ключа может привести к компрометации личности подписанта и к возможности подделки подписи.

В настоящее время для решения данной проблемы применяются решения класса «Доверенный платформенный модуль». Однако, подобные решения не могут удовлетворять базовым требованиям безопасности, поскольку подобные «модули» можно без труда извлечь из оконечного устройства и подключить к другому устройству. Более того, данное решение не подходит к массовому использованию по причине низкого уровня локализации.

Для предотвращения возникающих рисков, в том числе, в критической информационной инфраструктуре, необходимо обеспечить возможность хранения секретных ключей на оконечных устройствах без возможности их эффективного извлечения. Для этого предлагается внести изменения в Исходный код ядра операционной системы «Линукс» для электронно-вычислительных машин, предлагаемые настоящим Проектом.

Изменения предусматривают внедрение локального хранилища данных непосредственно в область памяти ядра операционной системы «Линукс», которое будет передавать данные данные с учётом требований информационной безопасности соответствующего лица, отвечающего за введение требований по информационной безопасности эксплуатанта.

Особое внимание в Проекте уделено вопросам обеспечения консистентности данных. Так, хранимые данные будут неизменны.

Принятие настоящего Проекта положительно повлияет на безопасность критической информационной инфраструктуры и отдельных устройств и программно-аппаратных комплексов, разработанных на базе решений ядра операционной системы «Линукс».

ФИНАНСОВО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ

к проекту фиксации «О внесении изменений в Исходный код ядра операционной системы «Линукс» для электронно-вычислительных машин»

Принятие документа «О внесении в порядке кодозаменительной инициативы в Список рассылки сопровождающих подсистем ядра проекта фиксации «О внесении изменений в Исходный код ядра операционной системы «Линукс» для электронно-вычислительных машин» не приведет к сокращению доходов.

Вместе с тем, принятие пункта 5 Постановления может повлечь за собой дополнительные расходы из средств бюджетных ассигнований в размере до 100,00 (ста) рублей на каждый программно-аппаратный комплекс на базе ядра операционной системы «Линукс» в качестве расходов на электроэнергию, необходимую для работы процесса сборки ядра операционной системы «Линукс».