OS Day ‘24

Secure-by-design vs hardenings

Нужны ли бинарные митигации безопасным операционным системам?

Анна Мелехова,
руководитель группы разработки защитных решений безопасной платформы,
«Лаборатория Касперского»

KasperskyOS

здесь была бы уместна моя фото если бы вы не видели меня вживую

Анна
20+ лет в ИТ

Виртуализация, архитектура, микроядро

Как харденилось ядро

Проблемы безопасности и зачем харденинги в «обычной» ОС

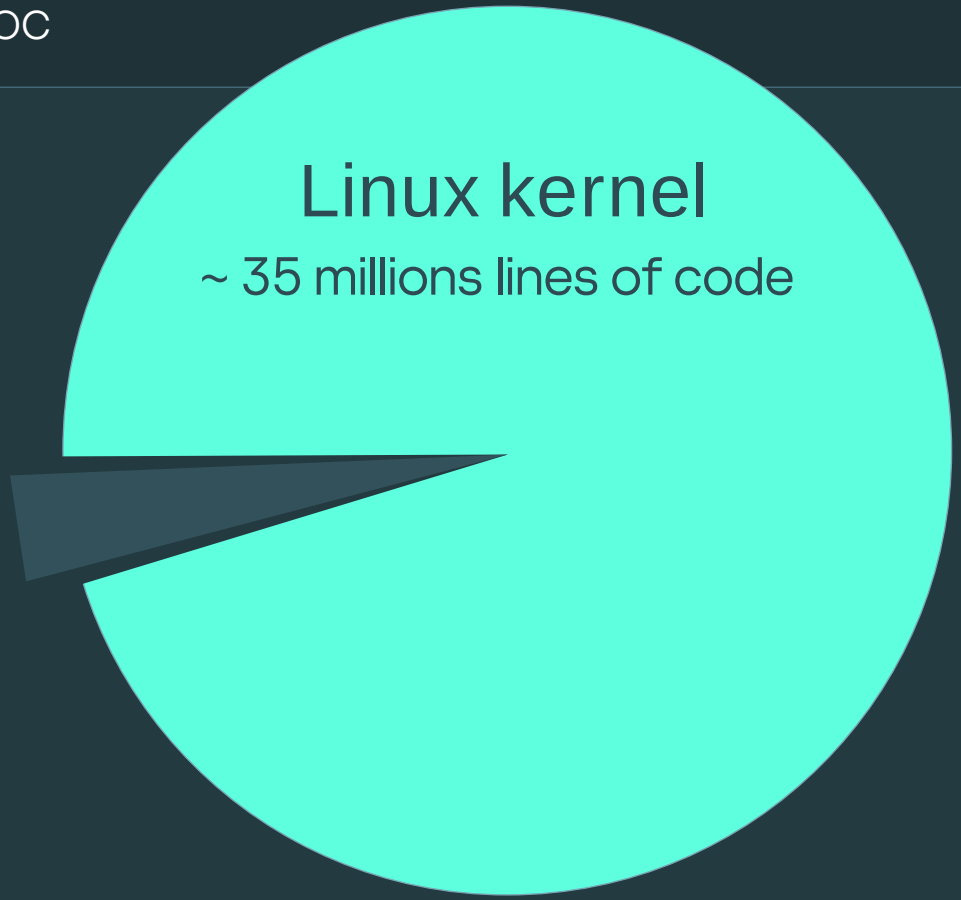
Базовые харденинги в Kaspersky OS

Супер-харденинги

Почему Linux'у сложно быть безопасным

И чем помогают харденинги

Minimizing the TCB

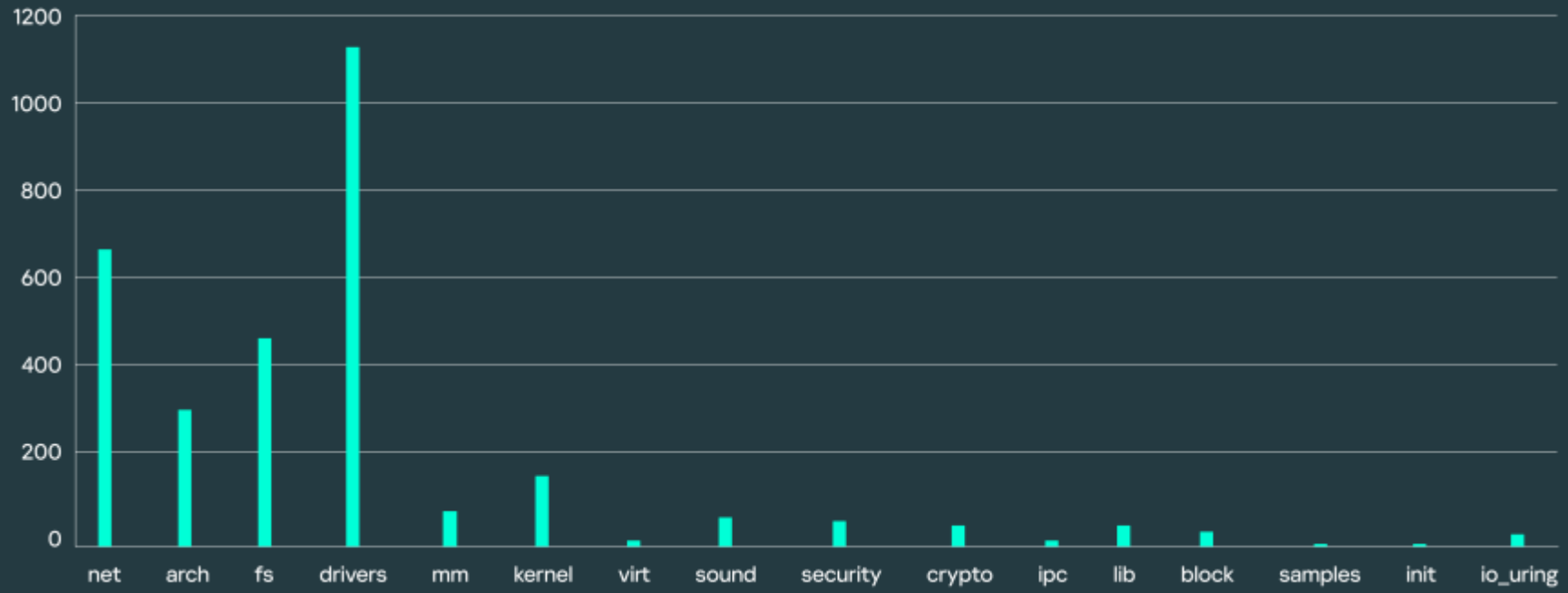


Microkernel

~ tens of thousands lines of code

Linux: атаки через драйвера

За всю историю ядра линукс CVE связаны с уязвимостями в драйверах **в 37% случаев**

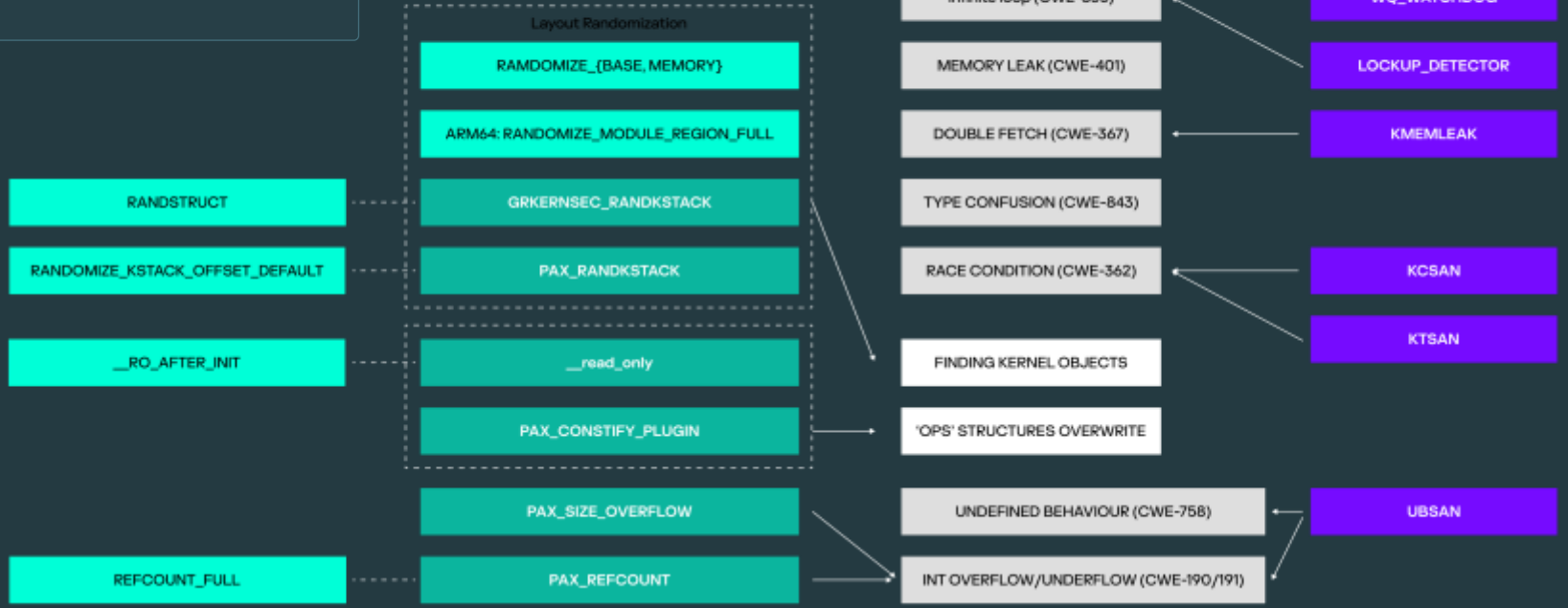


Linux: секреты

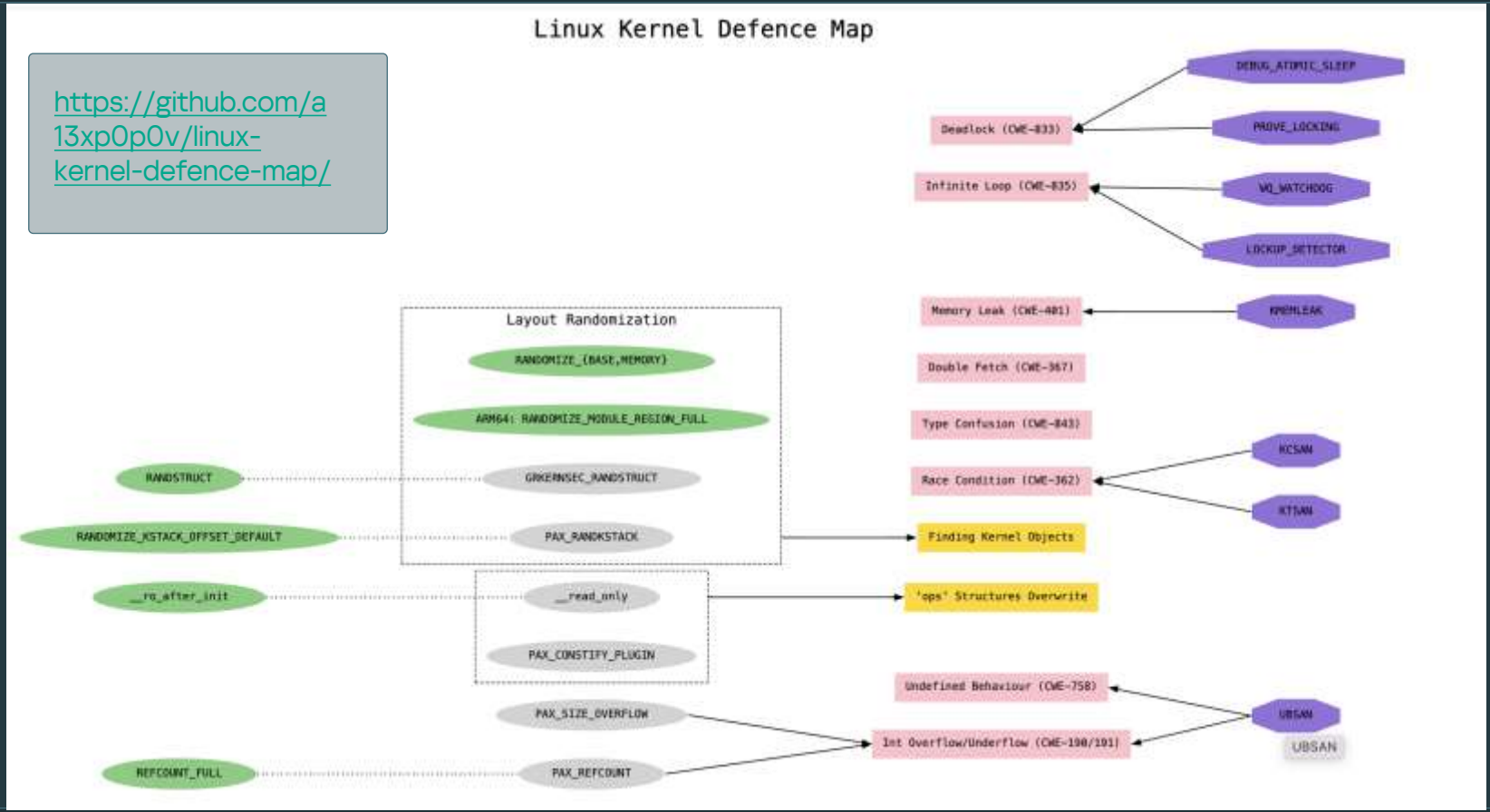
- Crypto API keys
- Random entropy pool
- Keyring: keyctl
- Ipsec keys
- Wireguard keys
- wifi keys
- KASLR

Linux kernel defence map by Alex Popov

https://github.com/a13xp0p0v/linux-kernel-defence-map/



https://github.com/a13xp0p0v/linux-kernel-defence-map/



Secure OS by Trent Jaeger

Complete Mediation: How does the reference monitor interface ensure that all security- sensitive operations are mediated correctly?

Complete Mediation: Does the reference monitor interface mediate security-sensitive operations on all system resources?

Complete Mediation: How do we verify that the reference monitor interface provides complete mediation?

Secure OS by Trent Jaeger (2)

- **Tamperproof**: How does the system protect the reference monitor, including its protection system, from modification?
- **Tamperproof**: Does the system's protection system protect the trusted computing base programs?
- **Verifiable**: What is basis for the correctness of the system's TCB?
- **Verifiable:** Does the protection system enforce the system's security goals?

> ...the combination of the Linux kernel and LSM framework is too complex for a complete formal verification that would be required to prove complete mediation and tamperproofing.
>
> ... The SELinux approaches demonstrates the complexity of UNIX systems and the difficulty in enforcing comprehensive security. The outstanding challenge is the definition and verification of desirable security goals in these low-level policies.

Trent Jaeger on Secure Linux

> Linux is not a secure operating system. However, there are steps you can take to improve it.

https://madaidans-insecurities.github.io/guides/linux-hardening.html

Традиционные бинарные митигации в ядре

Чем харденится KasperskyOS

Микроядро

Монитор безопасности, сгенерированный из политик безопасности и описаний интерфейсов компонентов

Secure by design

Контроль за аргументами: типизация и арена

Ограничение поверхности атаки – разрешенные вызовы

Генерируемый монитор безопасности

Микроядро

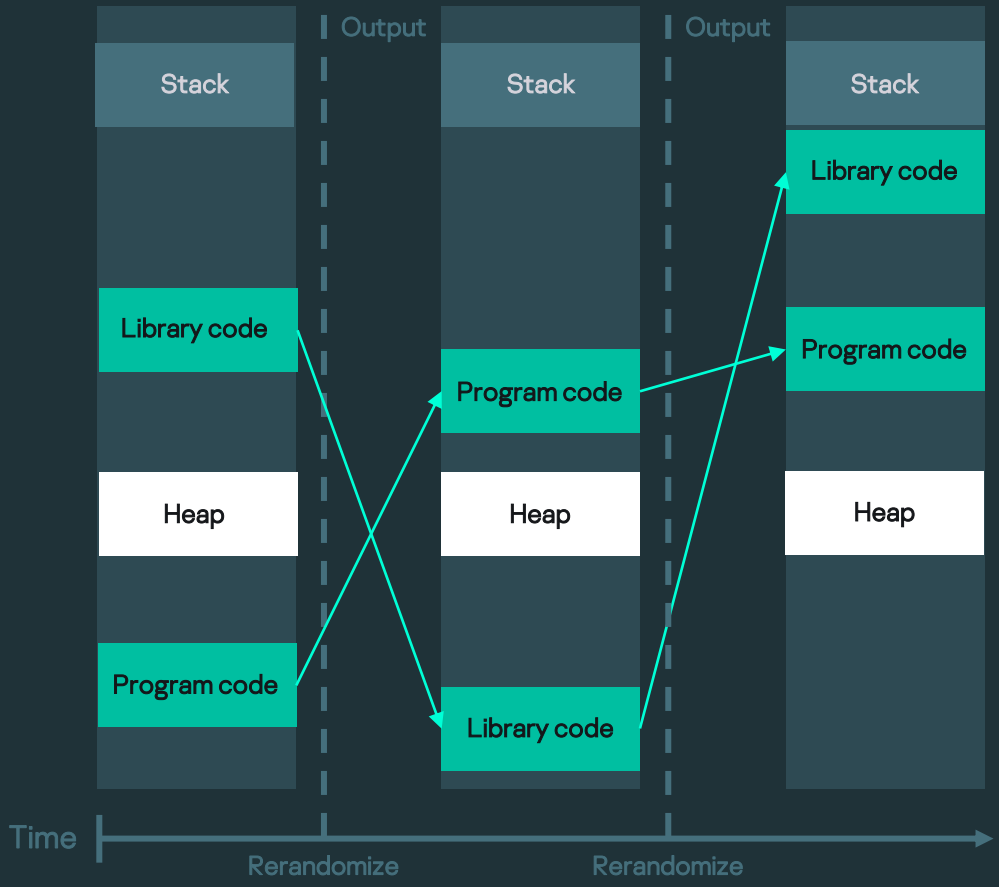
- Сторонний загружаемый код (драйвера) вне ядра
- Сокращение TCB
- Секреты, находящиеся вне ядра

Традиционные митигации

Базовые харденинги и все ли они нужны

- Стековая канарейка (stack canary)
- Рандомизация адресов (KASLR)
- Трамплин (KPTI)

Рандомизация адресов - ASLR (address space layout randomization)



Многоликий ASLR

Рандомизация размещения исполняемого кода

- Основного бинарника
- Динамических модулей

Рандомизация размещения стеков потоков

Рандомизация кучи

- Рандомизация базового адреса сегмента кучи
- Рандомизация элементов кучи

Рандомизация всех сегментов программы

- Рандомизация размещения data/bss/ro
- Рандомизация mmap

Рандомизация элементов внутри структуры

Атаки, от которых защищает KASLR:

- ACE через построение ROP/COP/JOP chain
- Утечка данных

Микроядро

- Меньше ROP/COP/JOP gadget
- Меньше секретов

ROP gadgets: Kaspersky OS vs Linux

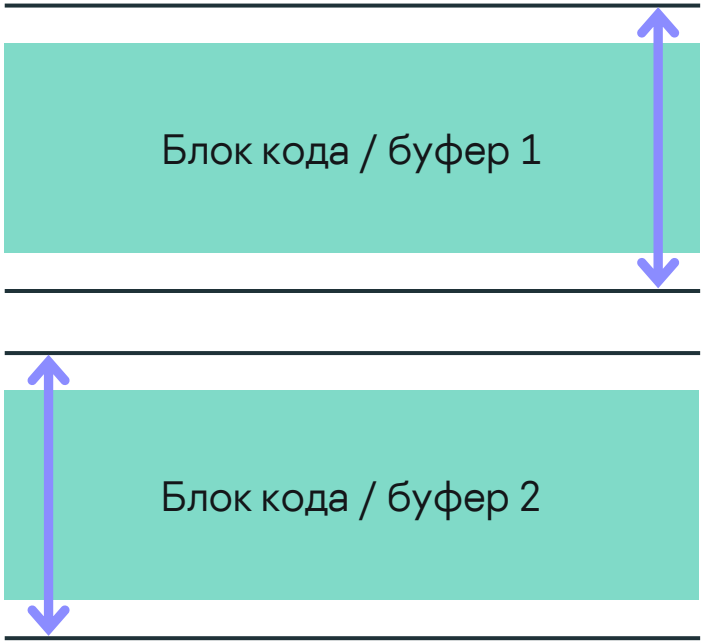

```
dev@dev-laptop:~/rop_test$ ropper --file vmkakos --section .text >
vmkakos.gadgets | tail -n 1
[INFO] Load gadgets for section: LOAD
[LOAD] loading... 100%
[LOAD] removing double gadgets... 100%
```

41293 gadgets found

```
dev@dev-laptop:~/rop_test$ ropper --file vmlinux --section .text >
vmlinux.gadgets | tail -n 1
[INFO] Load gadgets for section: LOAD
[LOAD] loading... 100%
[INFO] Load gadgets for section: LOAD
[LOAD] loading... 100%
[LOAD] removing double gadgets... 100%
220801 gadgets found
```

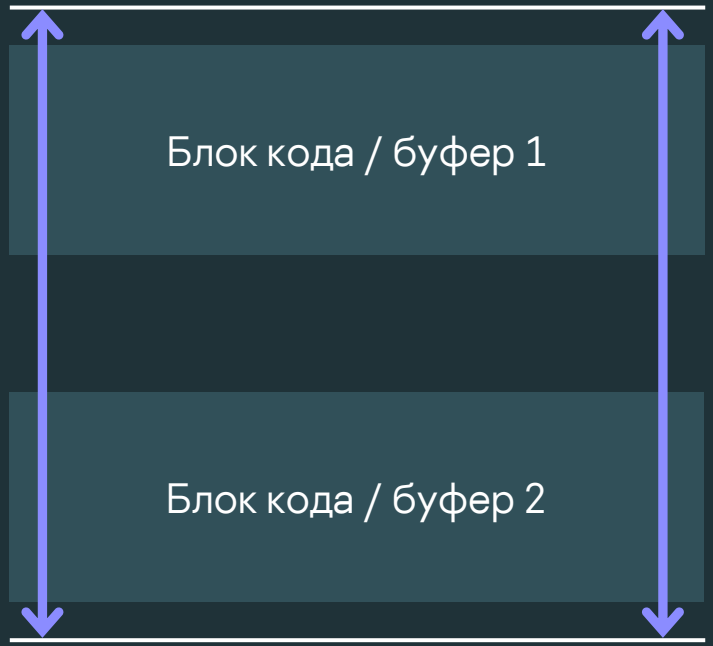
KASLR (Kernel address space randomization)

«Дрожание буферов»



KasperskyOS

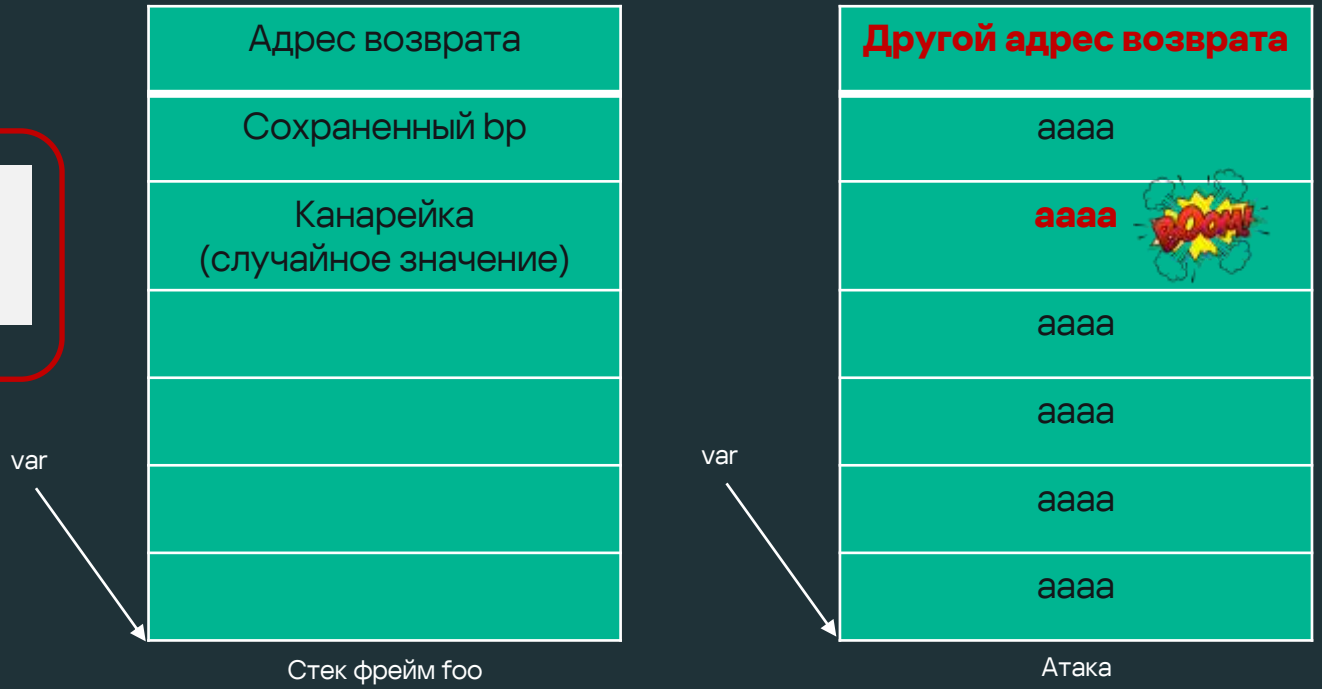
Полная рандомизация



Stack canary


```
void foo(){
  char var[12];
  scanf("%s", var);
}
```

Уязвимый код

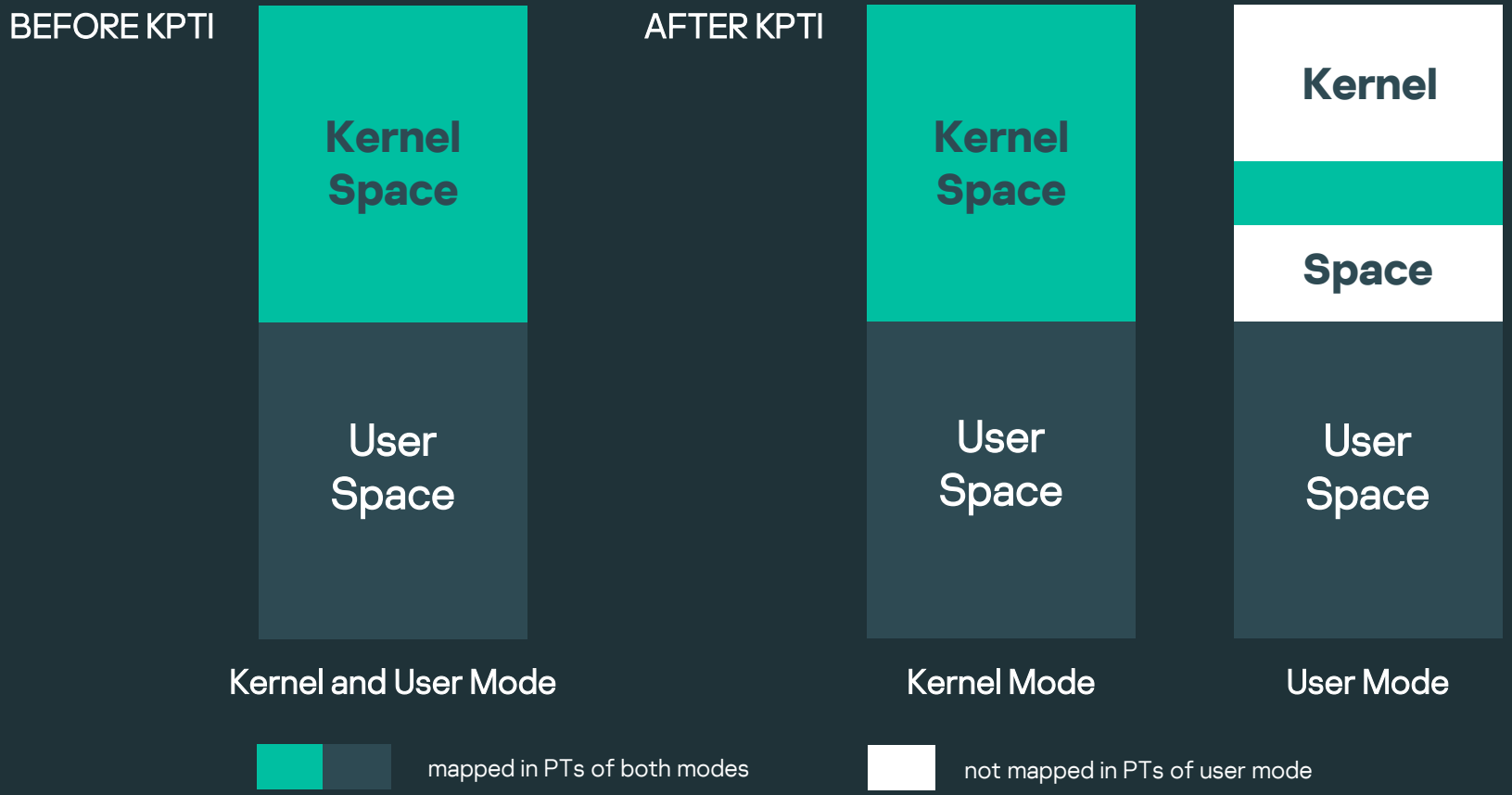


Stack canary

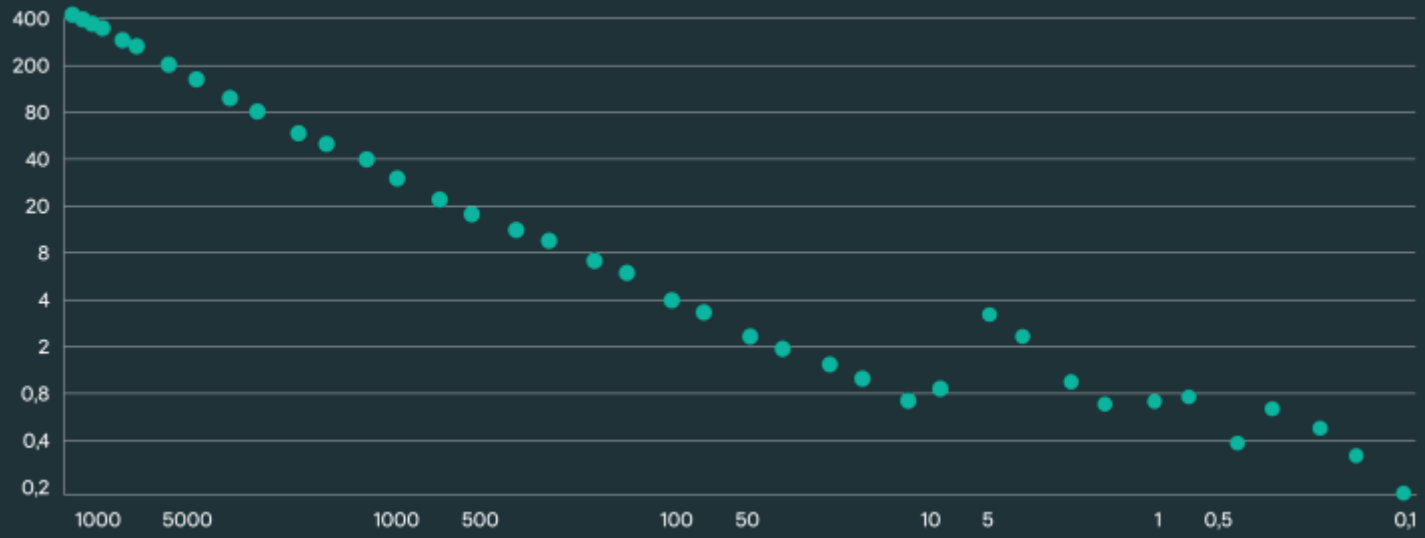
	Перезапись	**Чтение**	**Угадывание(fork)**
Windows	Секция data ✅	xor с ebp ✅	нет fork ✅
OpenBSD	RO секция ✅	xor с ret адресом ✅	❌
Glibc(Linux)	TCB/ ❌ data.rel.ro ✅	❌	❌
KasperskyOS	data.rel.ro ✅	xor с ebp ✅	нет fork ✅

Для множества архитектур:
x86, aarch64, arm

Трамплин (KPTI – Kernel page table isolation)



KPTI и performance downgrade



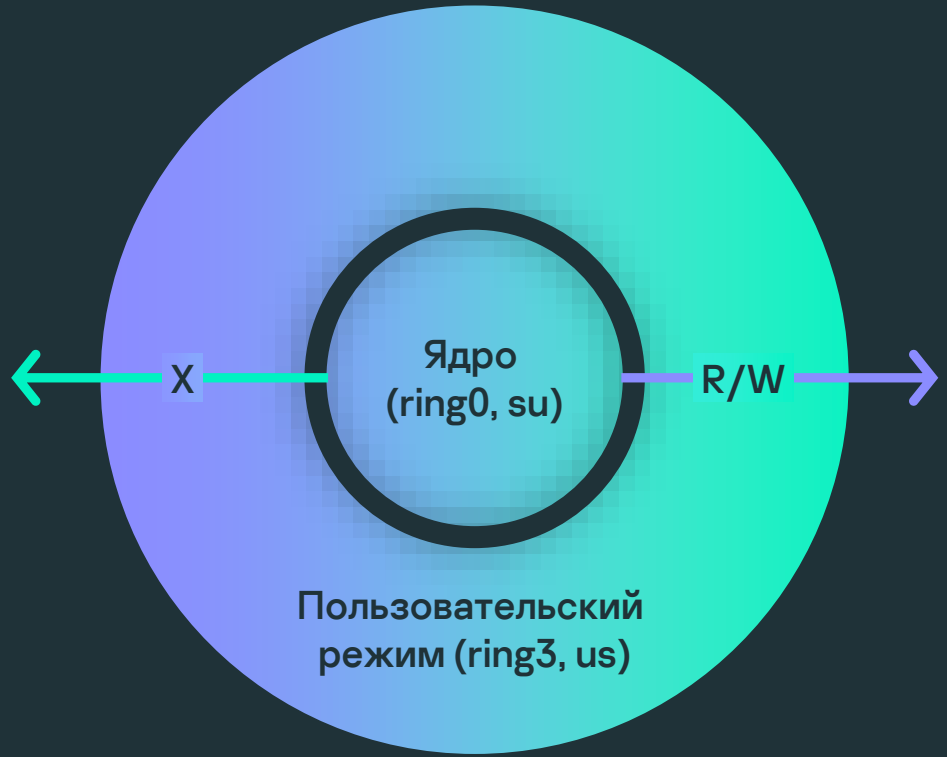
https://www.brendangregg.com/blog/2018-02-09/kpti-kaiser-meltdown-performance.html

> *KAISER will affect performance for anything that does system calls or interrupts: everything. ... Most workloads that we have run show single-digit regressions. 5% is a good round number for what is typical.*

https://lwn.net/Articles/738975/

У нас нет секретов

PXN/SMEP: нельзя просто так взять и исполнить код в user-space страницах памяти



PAN/SMAP: нельзя просто так взять и прочитать/записать данные в user-space страницах

Расширенные бинарные митигации

Что хорошо харденится в микроядре Kaspersky OS

W^X проблема

Страницы одновременно writable+executable идеальны для payload'а

Давайте не будем разрешать создавать W+X страницы?

- приложениям → Нет такой capabilities и не меняем default-ы
- и ядру с драйверами → Но как же BPF?

KasperskyOS

Давайте не будем разрешать W+X страницы?

- приложениям → А запретим! Разрешим лишь отдельным программам, например с JIT, через политики
- и ядру с драйверами → А запретим! У нас драйверы — в user-space, а ядро — микро

Выводы

Наложить безопасность на систему, изначальна для этого не спроектированную достаточно сложно

Микрокернел харденить нужно, хотя не все харденинги нужны

Опирайтесь на модель угроз и выбирайте свои харденинги для своих ОС

Спасибо!

А мы — молодцы ☺

Анна Мелехова

Руководитель группы разработки защитных решений безопасной платформы

Anna.Melekhova@kaspersky.com

KasperskyOS