Mecanismos anti-exploits em sistemas Linux

RONER DE CASTRO RODRIGUES

ESTUDANTE DE ENG. COMPUTAÇÃO - UNIPAMPA (BAGÉ)

DELPHI + SQL DEV. @ COBRAZIL.COM.BR

Tópicos

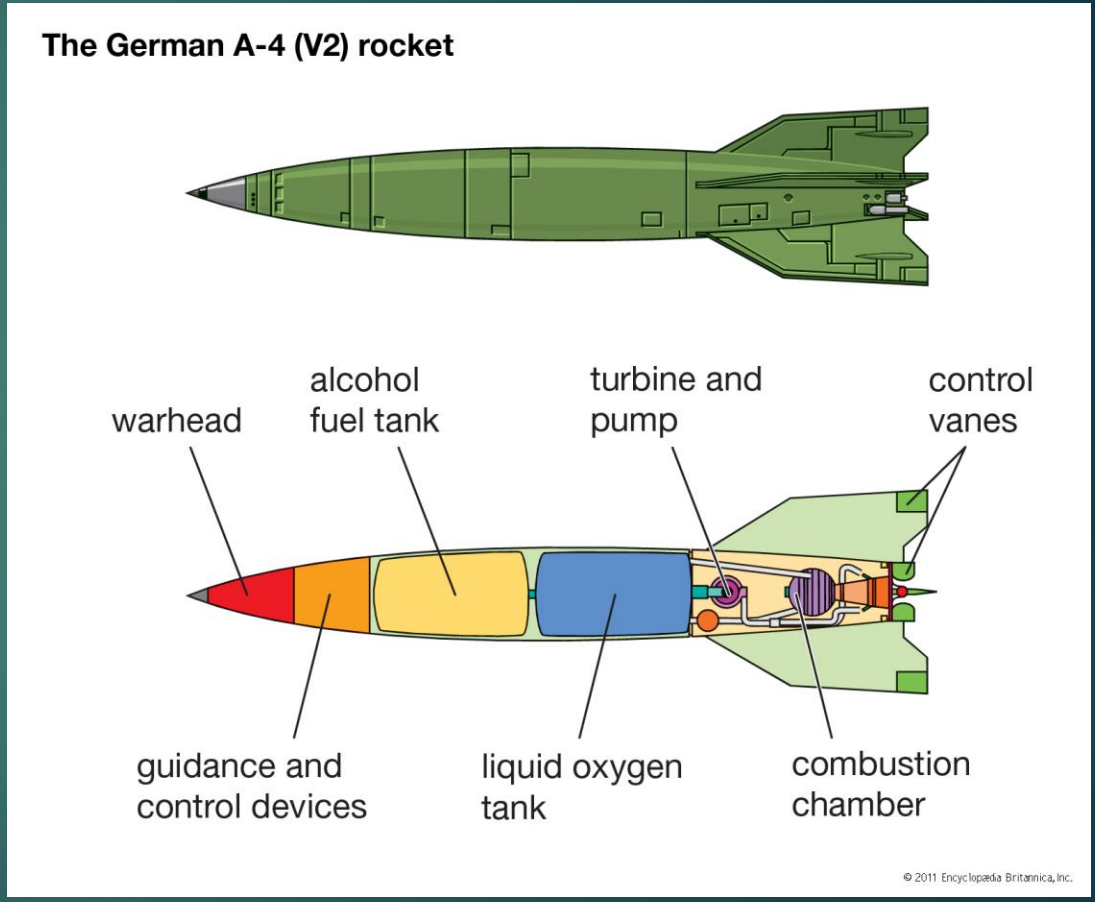
- O que são exploits ?
  - Classificação e funcionamento
  - Exploits famosos
- Alguns mecanismos de mitigação
  - Non Executable Bit (NX / XD)
  - ASLR / KASLR
  - Stack Canary
- Google Project Zero
  - Execução Especulativa
  - Meltdown
  - KPTI

O que são exploits ?

- Exploit é o pedaço de código / técnica responsável por forçar o sistema a fazer o que não devia:
  - stack / heap overflow
  - ret-to-lib / ret-to-dll
  - ROP Chain
  - Null pointer dereference
  - Falhas web - SQLi, RFI, XSS, etc
- Geralmente atuam nos mesmos lugares: stack, heap, handlers de exceção
- Aplicações diversas
  - Escalação de privilégios
  - Negação de serviço (DoS)
  - Hacking de jogos online (dll / code injection)
  - Vazamento de informações
- O maior problema sempre vai ser o dia-zero - não dá para evitar !!

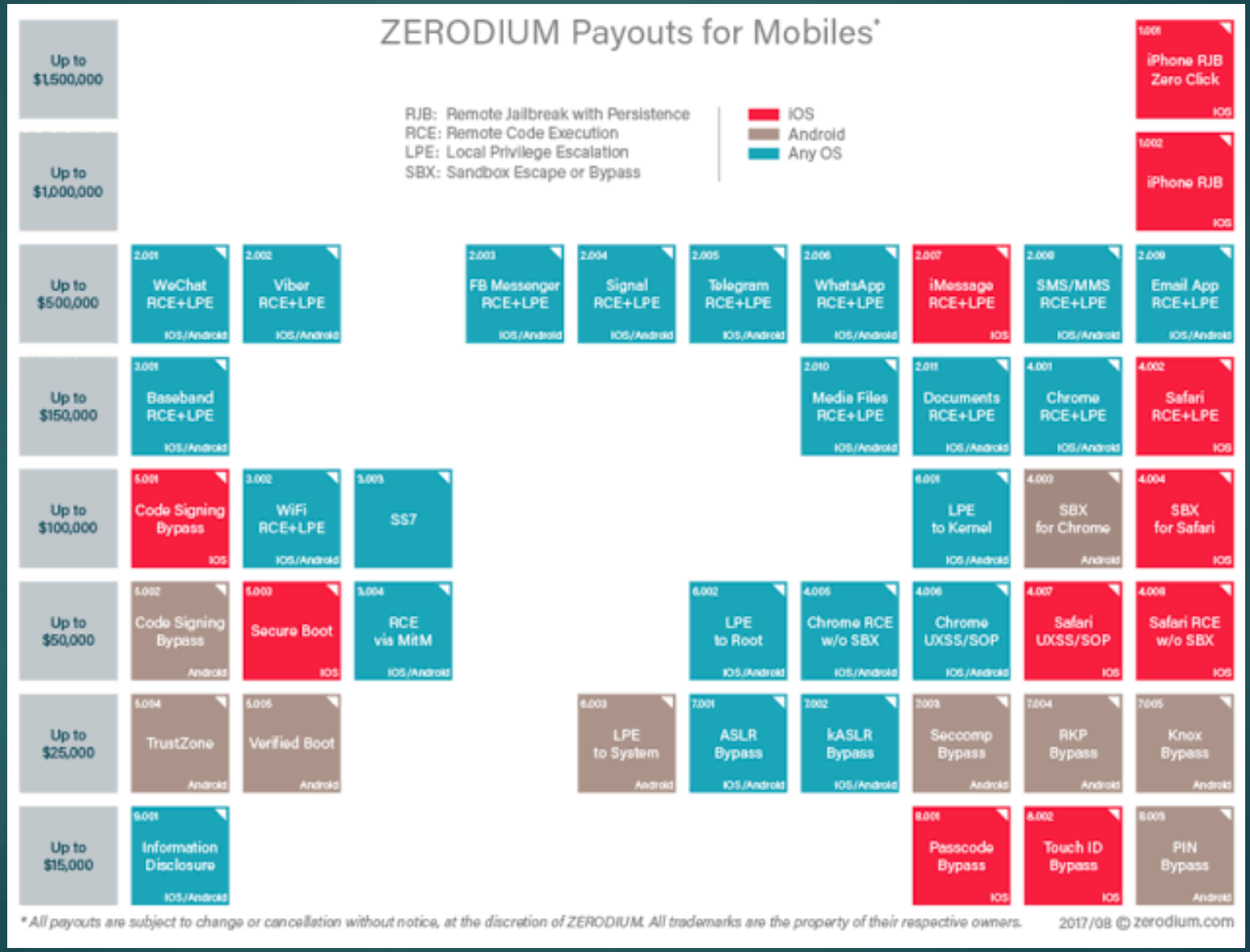
Funcionamento

- Para alguns autores, exploits consistem basicamente de 2 estágios:
  - **1.** Exploit – Aproveita-se de uma vulnerabilidade para (geralmente) desviar o IP para uma área de memória contendo código injetado pelo atacante (payload)
  - **2.** Payload / Shellcode - Código que será executado com fins diversos: escalação de privilégio, negação de serviço, conexão reversa, etc



- Analogia clássica com um míssil

Valores para exploits 0-day

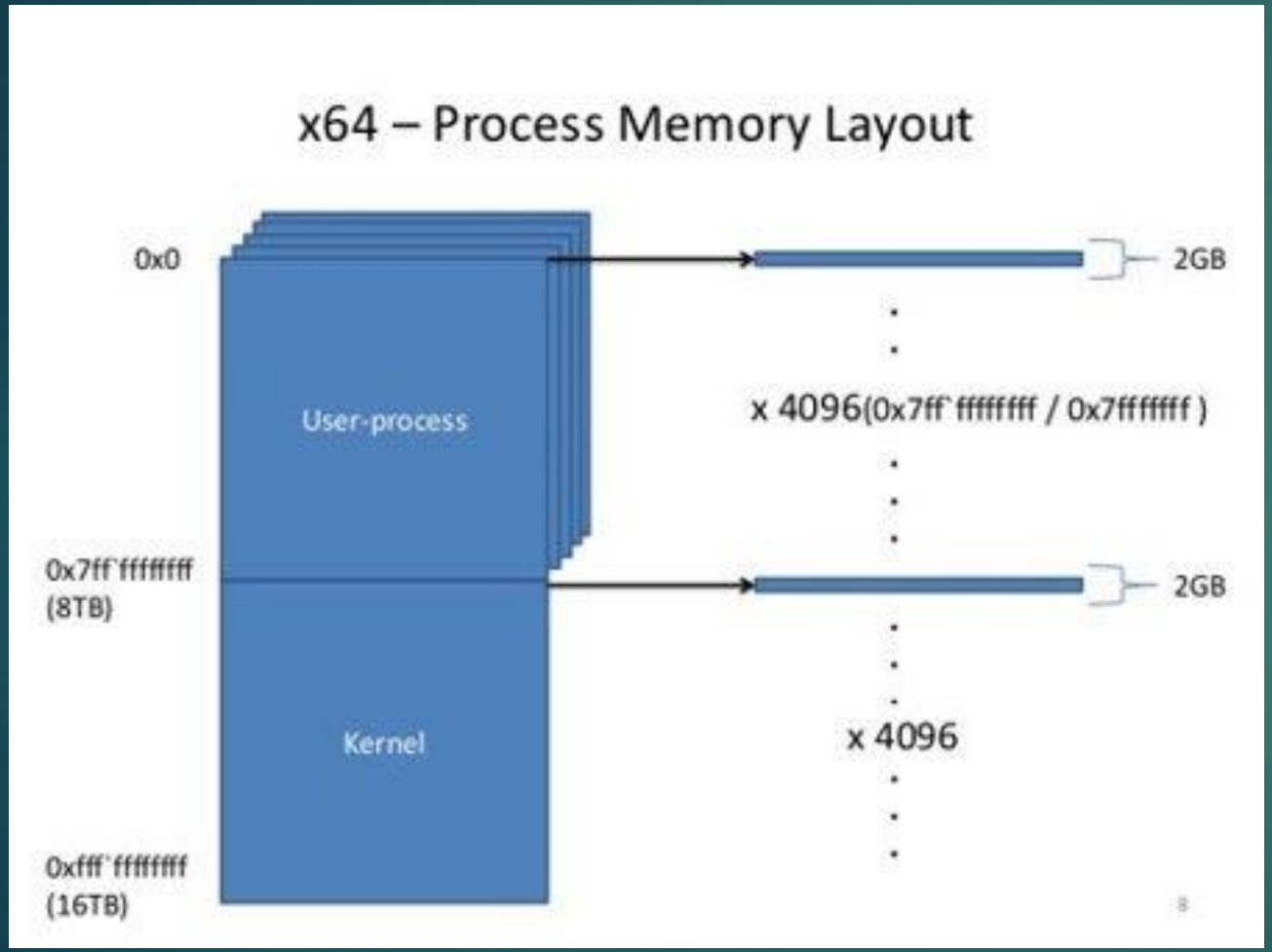


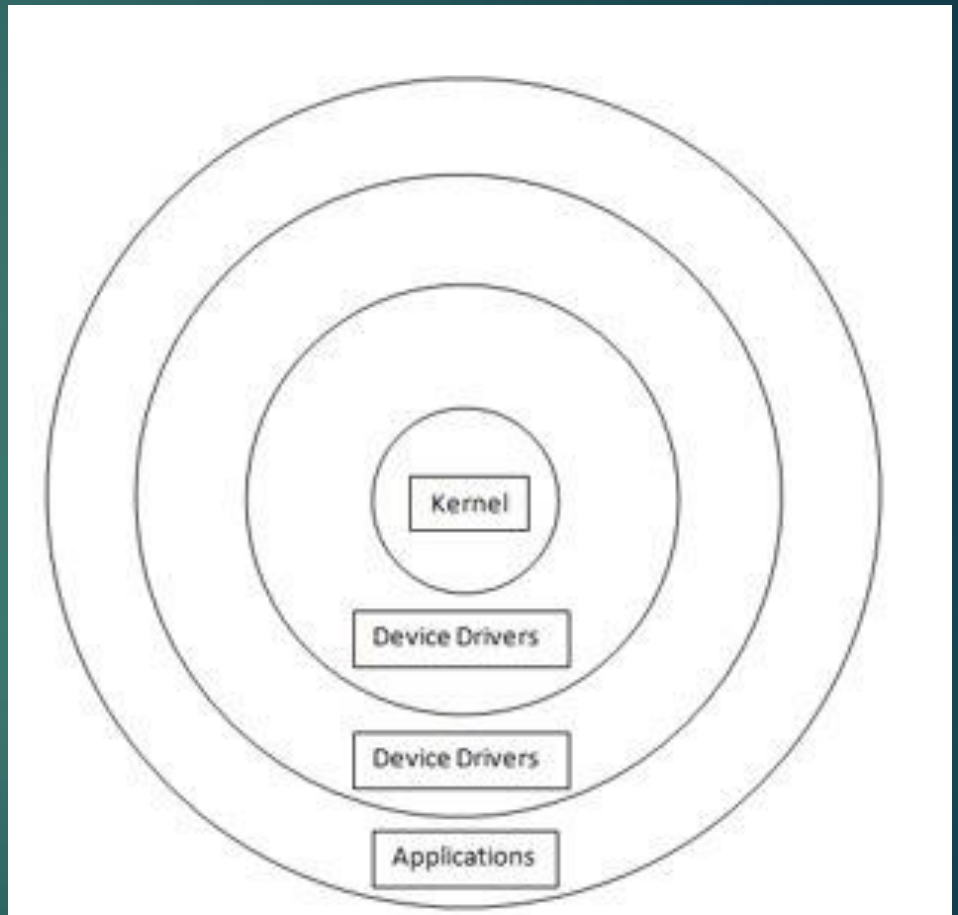
'Exploits famosos'

- **CVE-2010-2568** Uma das 4 falhas 0-day utilizadas no cyberataque contra o Irã no malware STUXNET. Consiste de código malicioso inserido em arquivos .LNK em drivers USB externos, que são lidos pelo Windows para exibição de ícones de atalho.
- **CVE-2017-5754** Meltdown – Vazamento de informação do kernel através do recurso de execução especulativa = quase 20 anos de processadores fabricados.
- **CVE-2017-5753** Spectre – Permite enganar programas 'livres de erro' e que seguem as 'melhores práticas'. Mais difícil de exploitar do que o Meltdown, porém mais difícil de mitigar também.
- **KRACK** Vulnerabilidade no handshake WPA2 em alguns dispositivos wifi que permite vazamento de informação. Estima-se que pelo menos 41% dos dispositivos Android são vulneráveis, assim como produtos Apple, Windows, OpenBSD e outros.
- **CVE-2017-0144** EternalBlue – cyberarma desenvolvida pela NSA para auxiliar em suas operações. Se aproveita de vulnerabilidade no protocolo SMBv1 que permite execução remota de código. Utilizada pelo WannaCry e ransonwares

Mecanismos de Mitigação

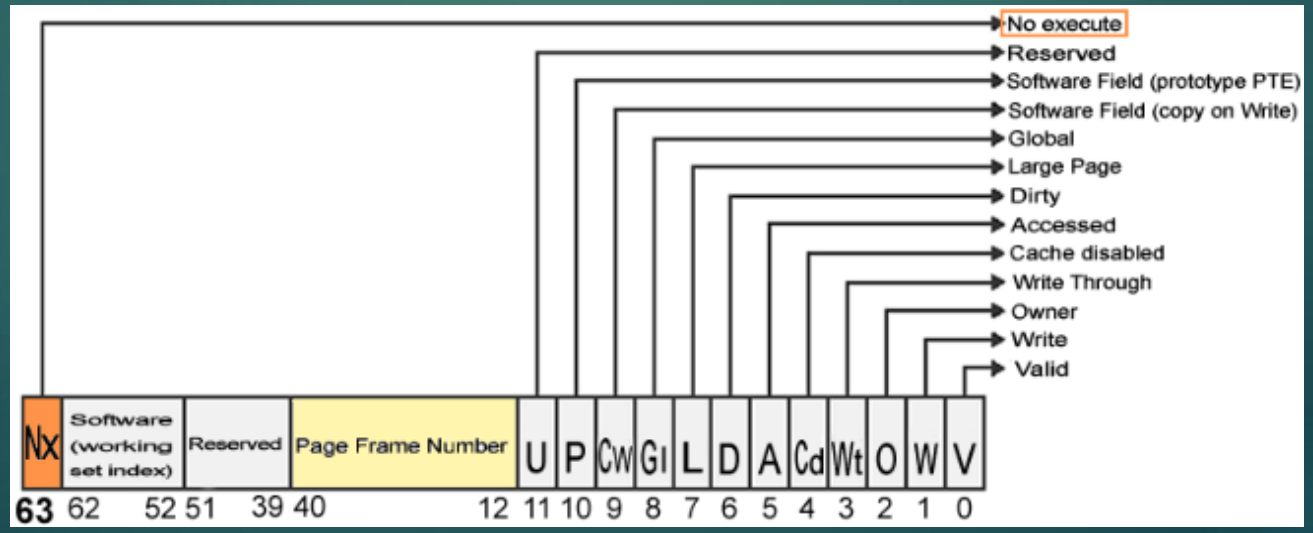
Layout de memória / Níveis de Privilégio





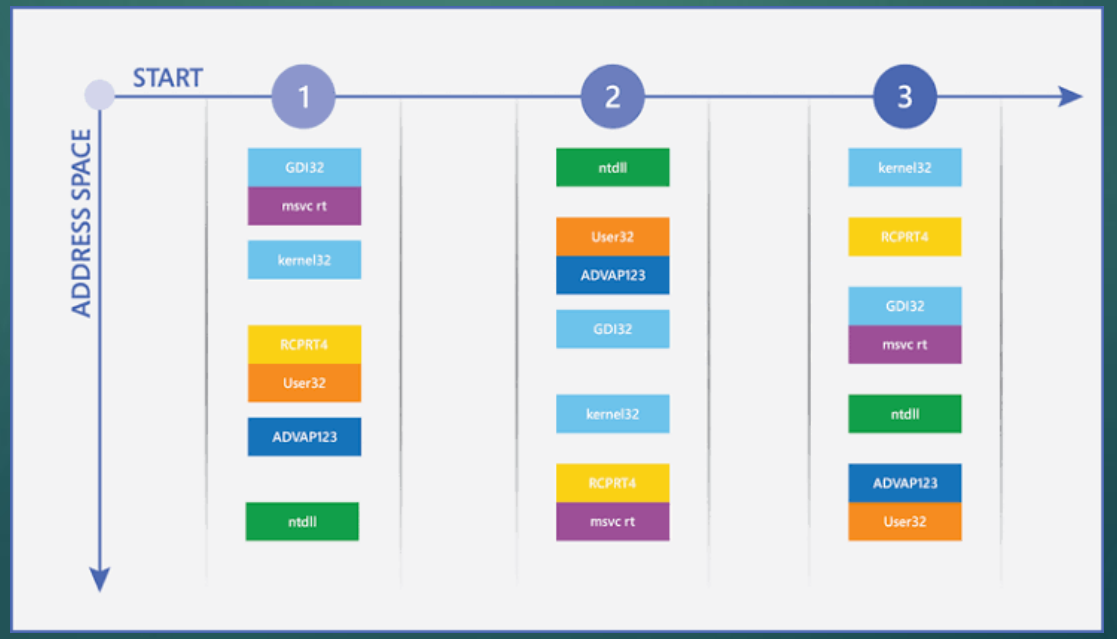
Non Executable Bit (NX / XD)

- Disponível a partir do AMD64 / Pentium 4 (Prescott) em diante (2003)
- Implementações: Writable xor Executable (W^X) / Data Execution Prevention (DEP)
- A partir do kernel 2.6.8, páginas do kernel podem ser marcadas com o flag **NX**
  - Funciona mesmo sem a disponibilidade do bit NX no processador (PAE)
  - Disponível através do PaX, ExecShield e já habilitado por padrão em algumas distros
  - Vulnerável a bypass via arc injection / ret-to-lib



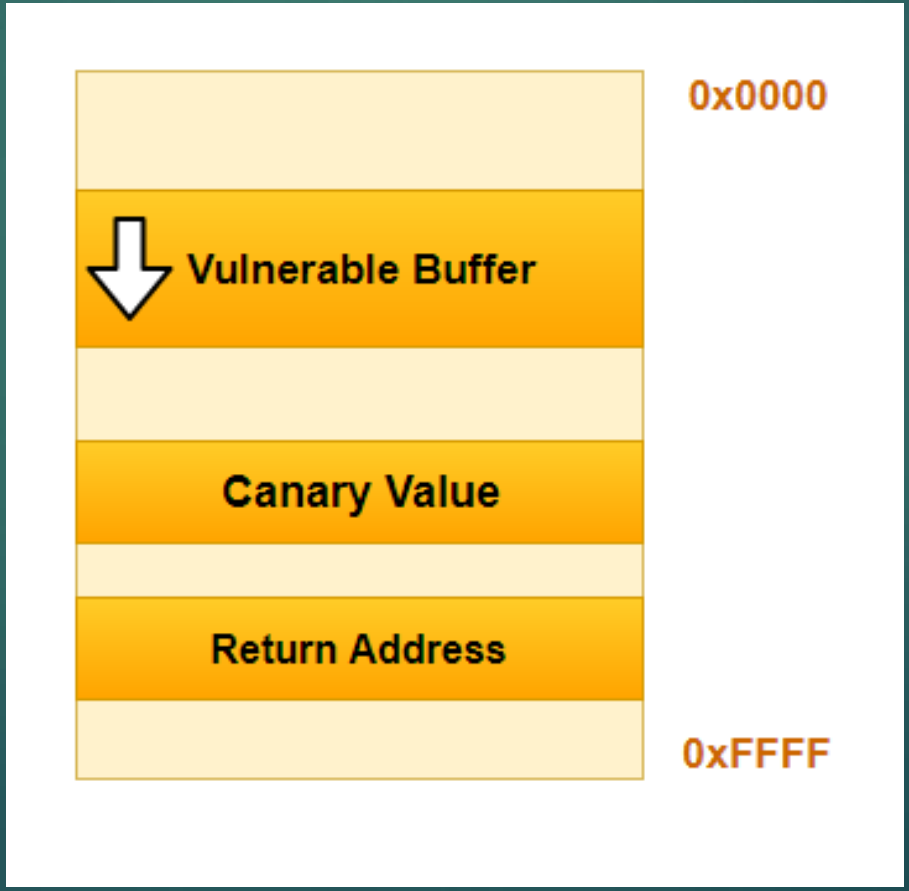
ASLR / KASLR

- Implementado por padrão no kernel 2.6+ / WinVista / WinServer 2008+
- Posição de segmentos e bibliotecas randomizados no momento da inicialização do programa / kernel
- Disponível no PaX – Problemas : baixa entropia, distribuição de probabilidade ruim, ...
- Evolui para ASLR-NG
- Vulnerável a bypass via memory leak, heap spray, etc



Stack Canary / StackGuard

- /GS flag: mitiga buffer overflows (x32) através da adição de uma checkum entre buffers e o endereço de retorno de suas respectivas funções;
- Mesmo assim... ainda é possível advinhar a checksum



E mais, muito mais...

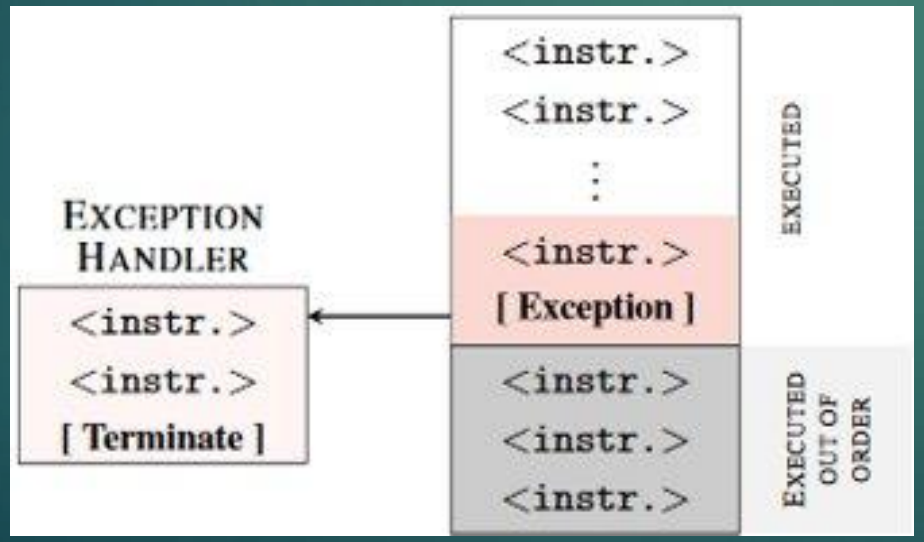
- Diversos projetos de segurança:
  - StackShield, StackGuard, Control Flow Integrity, PaX, Exec-Shield, OpenWall, GrSecurity, SELinux, checksec.sh ...
  - Mecanismos adicionais em distros fortalecidas: BSDs, Hardened Gentoo, etc
- ... que por sua vez implementam uma ou mais diferentes tecnologias
  - /SAFESEH + SEHOP (Windows)
  - Null-page protection
  - Position Independent Execulables / PIE
  - RELocation Read-Only / RELRO
  - Heap Corruption Detection
- Os principais mecanismos já vem implementados por padrão em distros mais modernas / atualizadas

Google Project Zer0

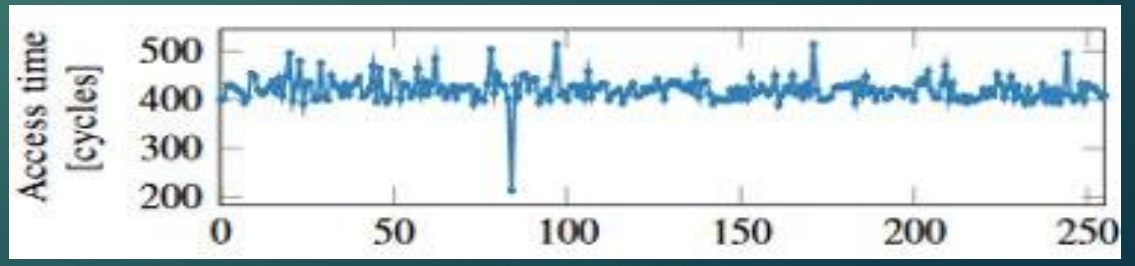
	Meltdown	Spectre
Allows kernel memory read	Yes	No
Was patched with KAISER/KPTI	Yes	No
Leaks arbitrary user memory	Yes	Yes
Could be executed remotely	Sometimes	Definitely
Most likely to impact	Kernel integrity	Browser memory
Practical attacks against	Intel	Intel, AMD, ARM

Meltdown (Variante 3)
Rogue Data Cache Load

- side-channel attack - vazamento de dados (tudo) através do user-mode via linhas da cache (L1D) = não é necessário exploitação / escalação de privilégios !
- Vale-se de recursos de execução especulativa / execução-fora-de-ordem
- Mais comum processadores Intel:TSX instructions – agrupamento 'tudo ou nada', logo não é necessário o uso de rotinas de handling de exceção

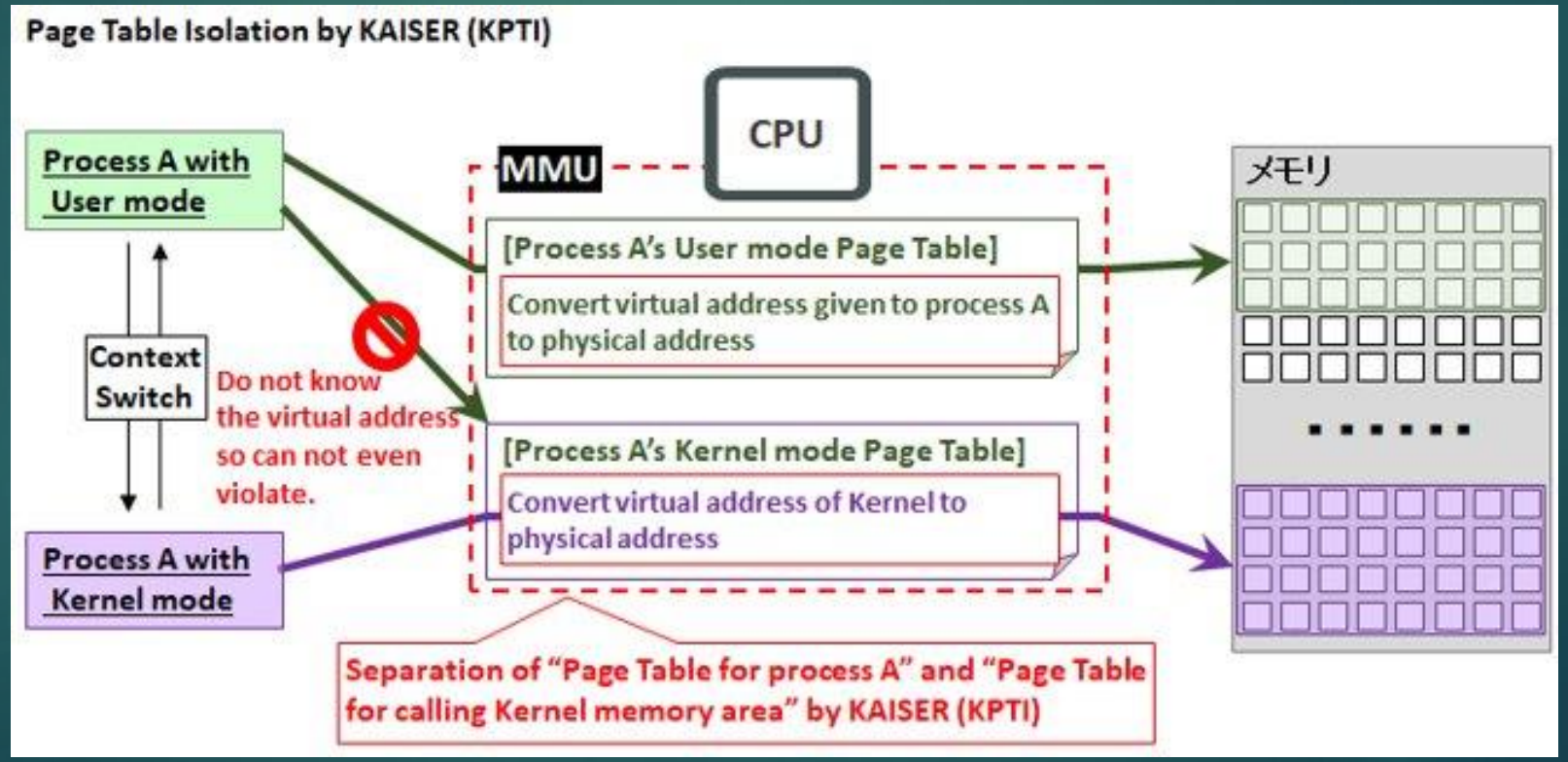



```
1 raise_exception();
2 // the line below is never reached
3 access(probe_array[data * 4096]);
```



KPTI - Kernel Page Table Isolation

- Isola o alcance e a visibilidade da superfície do kernel para o user-mode, através de uma troca de contexto forçada
- Processador agora vai trabalhar com 2 tabelas de páginas
- Overhead considerável ~= 30%



Muito Obrigado !