THE DEVELOPER'S CONFERENCE

SP15:Trilha Segurança

Globalcode

THE DEVELOPER'S CONFERENCE

SP15:Trilha Segurança

Locks are so old-fashioned…

Globalcode

THE DEVELOPER'S CONFERENCE

Anti-debugging: eu não quero que você mexa no meu código

Globalcode

THE DEVELOPER'S CONFERENCE

Wanderley Caloni

Sócio-Desenvolvedor da

Globalcode

THE DEVELOPER'S CONFERENCE

Wanderley Caloni

Sócio-Desenvolvedor da

<B1TF0RGE>

Globalcode

Agenda

THE
DEVELOPER'S
CONFERENCE

Globalcode

Agenda

THE DEVELOPER'S CONFERENCE

Jabá Time!

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

2013-2014-…

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

2013-2014-…

Prova incontestável de autenticidade!

Globalcode

Onde sou? Quem estou?

2000 e bolinha (??)

Onde sou? Quem estou?

THE
DEVELOPER'S
CONFERENCE

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

tempest

SECURITY INTELLIGENCE

open communications security

Globalcode

Onde sou? Quem estou?

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

UOL DIVEO

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

Exemplos de projetos/clientes da Intelitrader/BitForge:

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

Exemplos de projetos/clientes da Intelitrader/BitForge:

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

- Segurança da informação
- Mercado financeiro
- Software de baixo nível
- Sistemas críticos
- Linguagens
  - C, C++, .NET, VB6, Python, Delphi, Assembly, ASP.NET, SQL, HTML5, PostGres, Oracle, Inglês, Português, Russo, Polonês e todas as outras.

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

Google

INTELITRADER

<BITFORGE>

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

Microsoft

INTELITRADER

<BITFORGE>

Globalcode

Onde sou? Quem estou?

THE DEVELOPER'S CONFERENCE

INTELITRADER

<BitForge>

Globalcode

É isso aí pe-pe-pe-pe-pe…

THE DEVELOPER'S CONFERENCE

Jabá End

Globalcode

Agenda

THE DEVELOPER'S CONFERENCE

- Interpretação baseada em exceção
  - **int 3**
- Ocupando a debug port
  - **Debug Port**
- Detectando attach
  - **Attach**
- **Conclusão**

Globalcode

int 3

THE
DEVELOPER'S
CONFERENCE

?

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

```
int x = 3;
```

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

~~int x = 3;~~

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

$$\int_2^{\infty} \frac{1}{x \ln x}\,dx = \lim_{t\to\infty} \int_2^{t} \frac{1}{x \ln x}\,dx \qquad u = \ln x$$

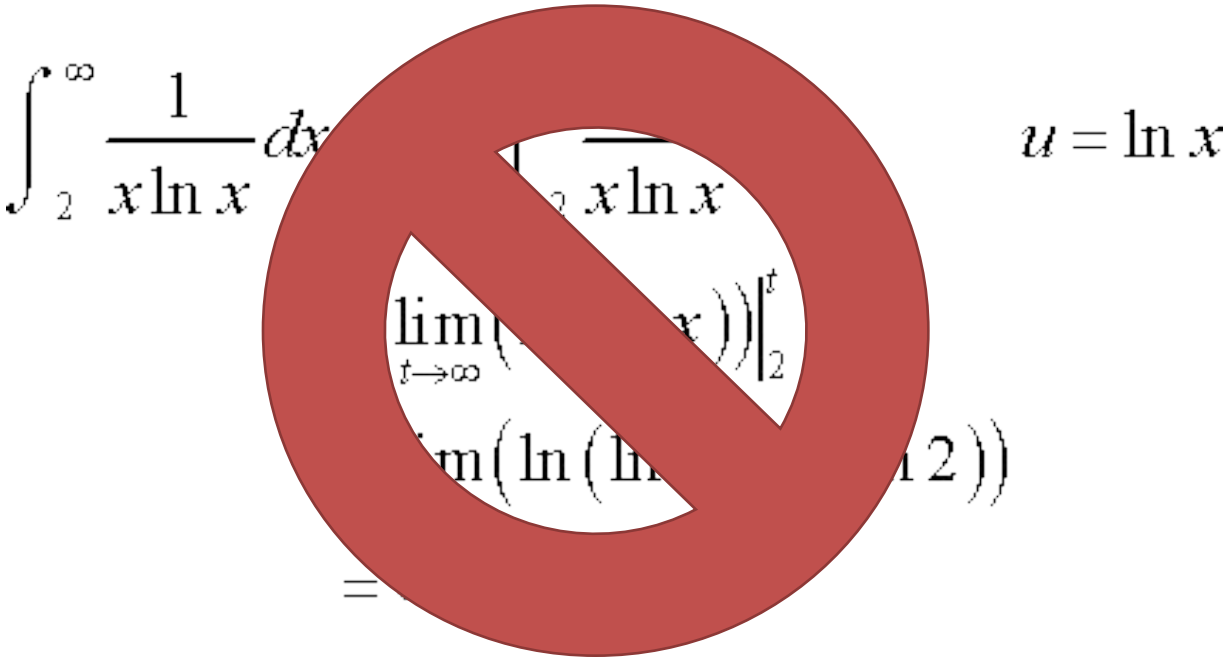
$$= \lim_{t\to\infty} \left(\ln\left(\ln x\right)\right)\Big|_2^t$$

$$= \lim_{t\to\infty} \left(\ln\left(\ln t\right) - \ln\left(\ln 2\right)\right)$$

$$= \infty$$

Globalcode

int 3



int 3

THE DEVELOPER'S CONFERENCE

asm

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

assembly

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

assembly

Globalcode

int 3

THE
DEVELOPER'S
CONFERENCE

assem

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

nop

nop

nop

nop

...

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

nop
nop
int 3
nop
...

Globalcode

int 3

```
nop
nop
int 3
nop
...
```

int 3

nop
nop
int 3
nop
...

int 3

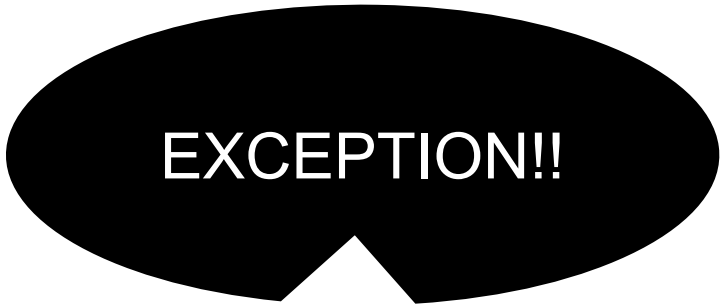
nop

nop

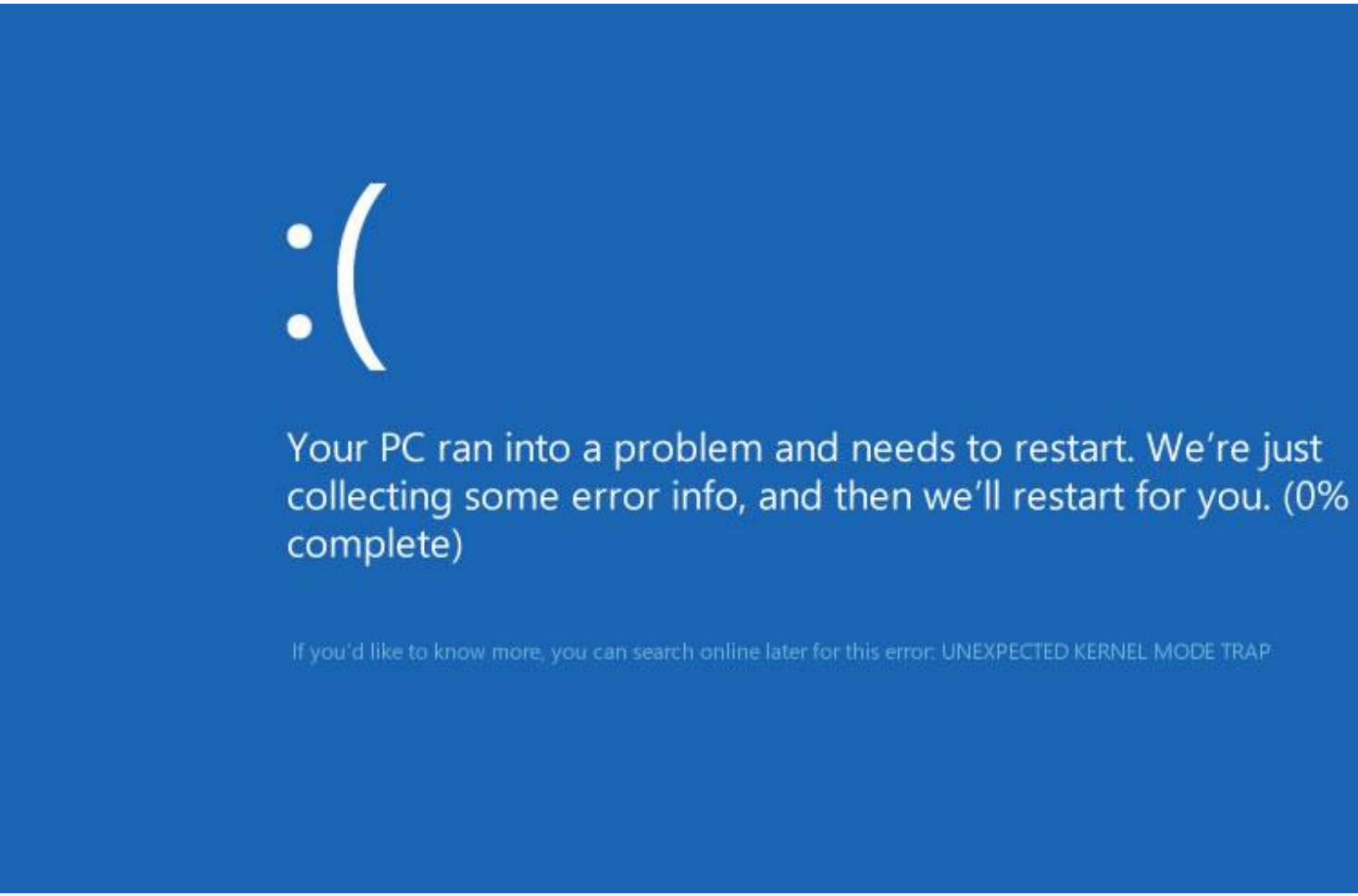
int 3

nop

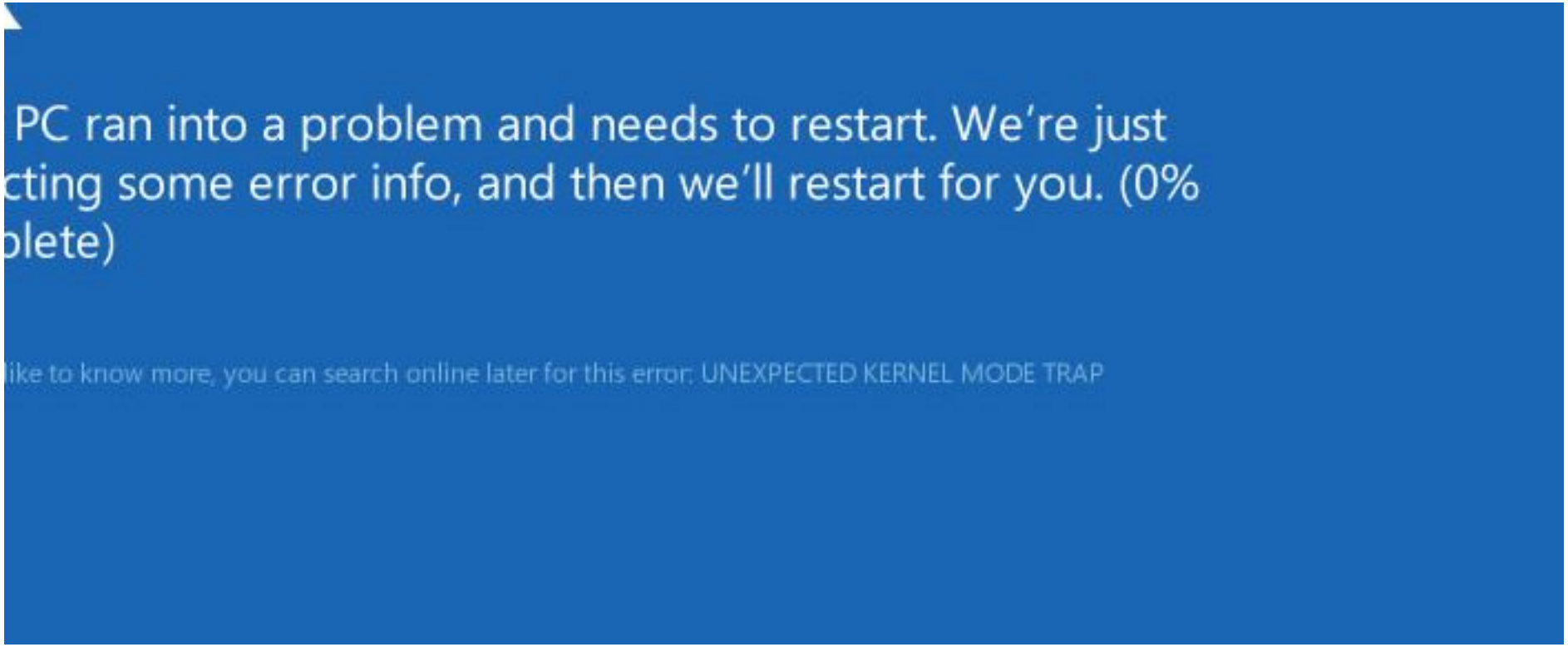
...



int 3



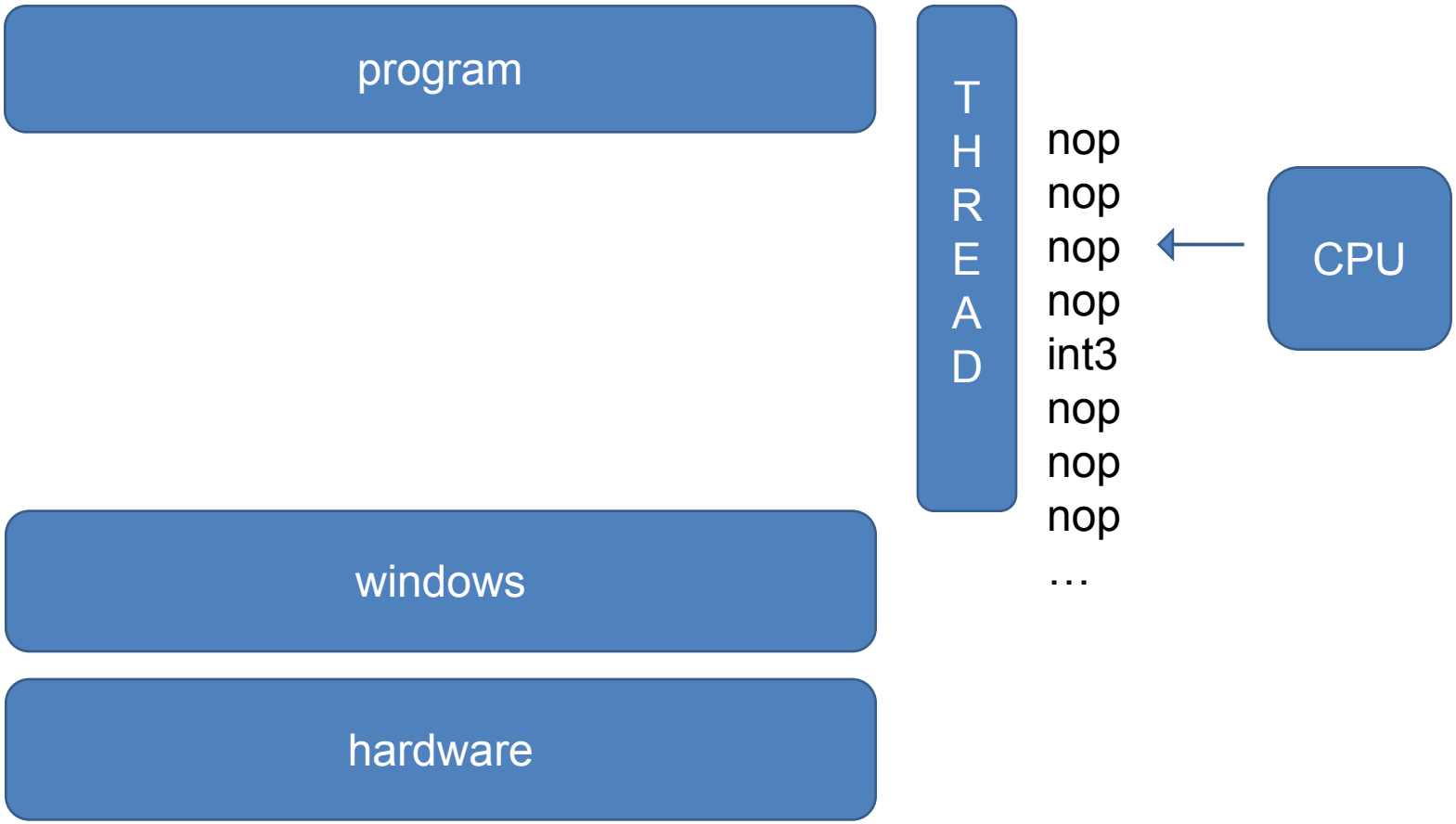
int 3



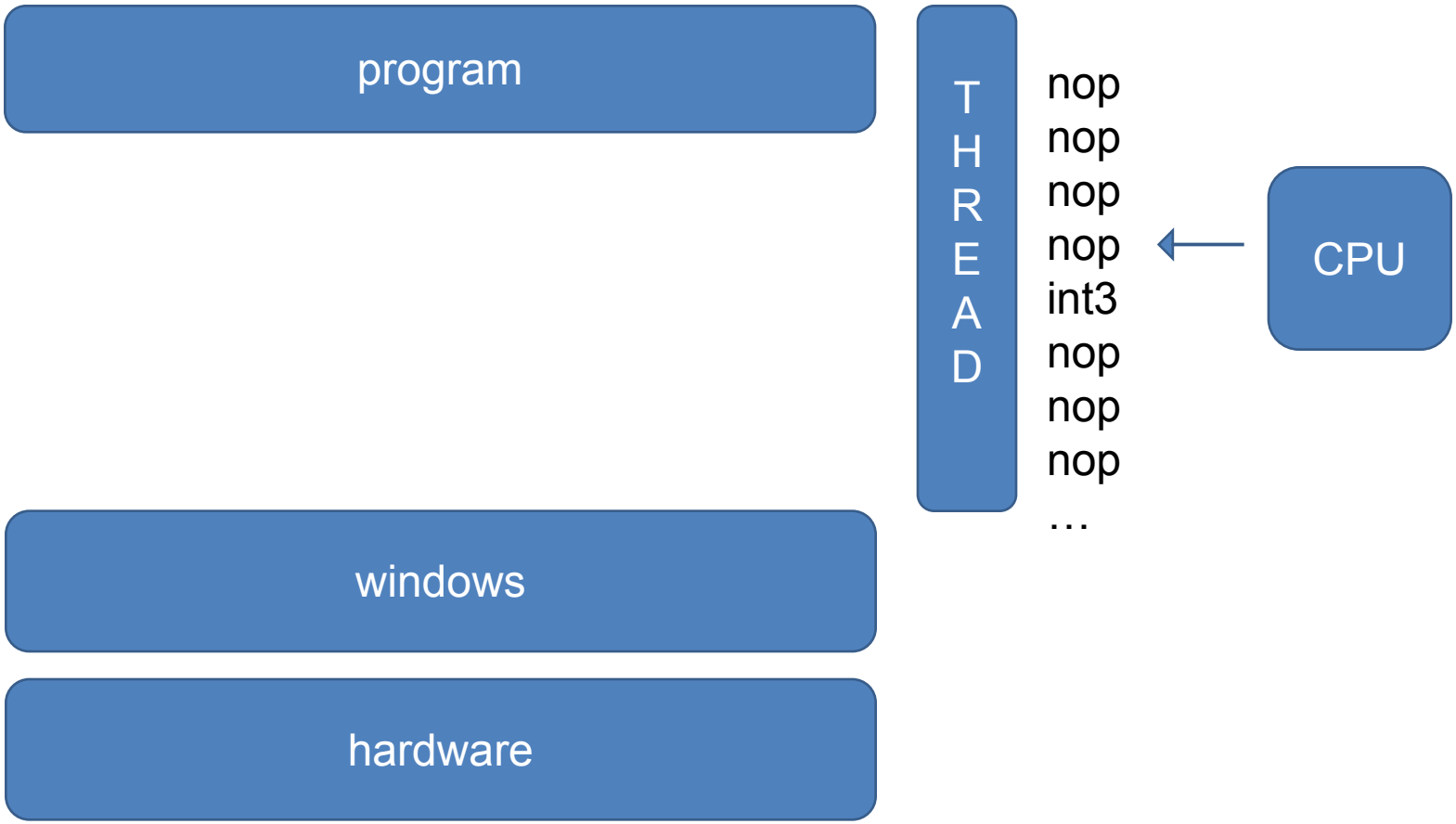
int 3

int 3

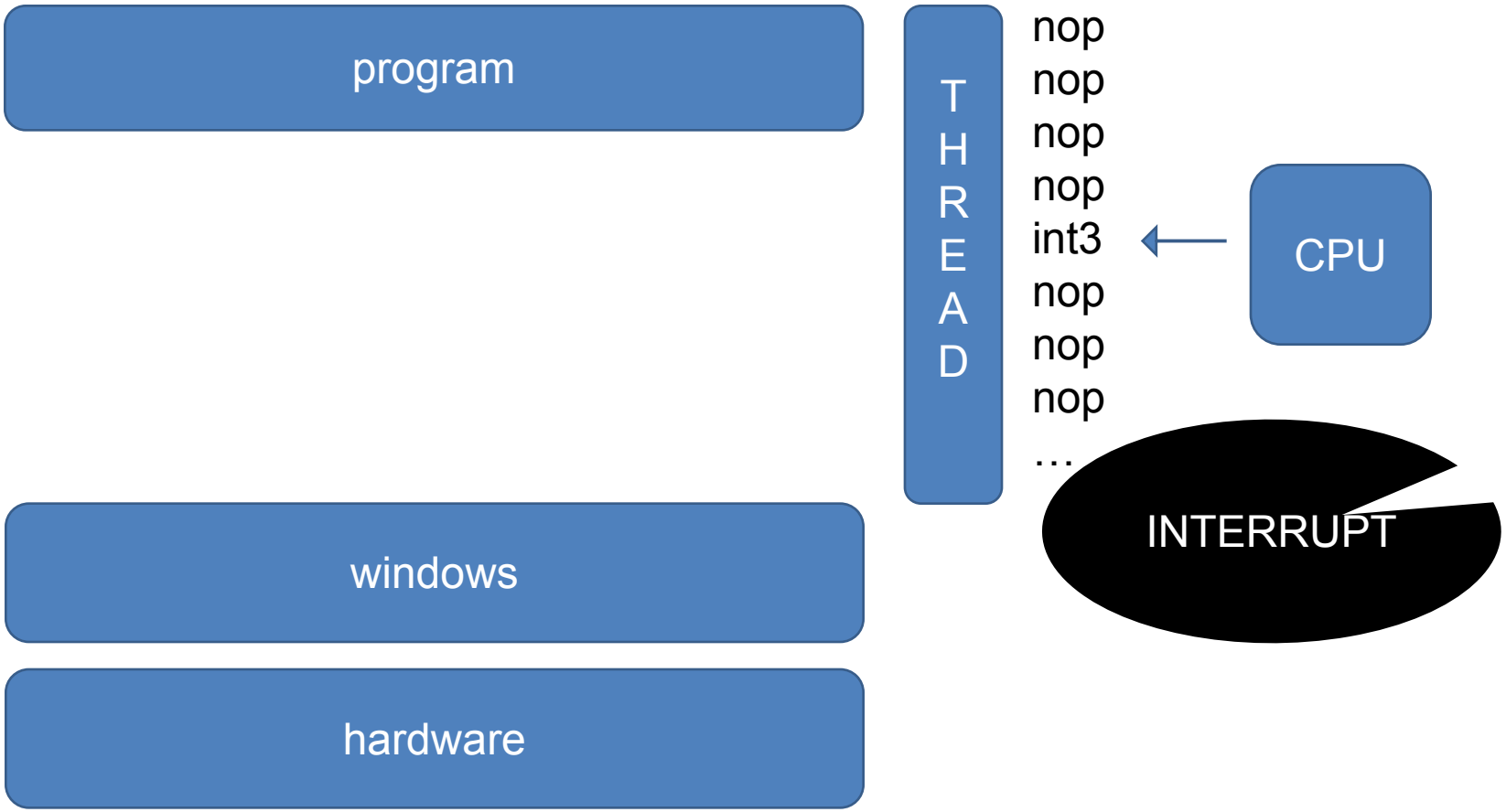
int 3


program
windows
hardware
THREAD
nop
nop
nop
nop
int3
nop
nop
nop
…
CPU

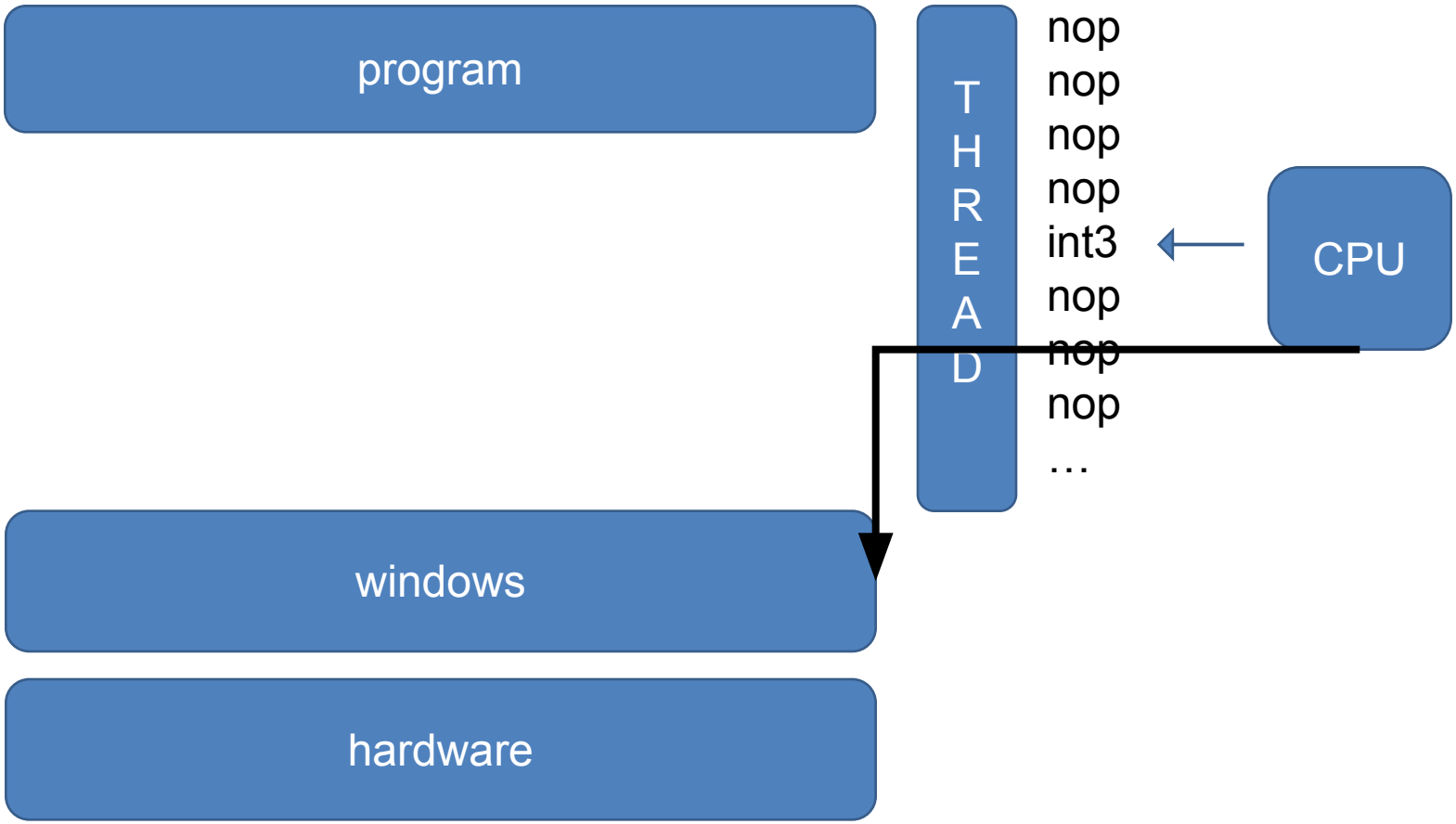
int 3


program
windows
hardware
THREAD
nop
nop
nop
nop
int3
nop
nop
nop
…
CPU

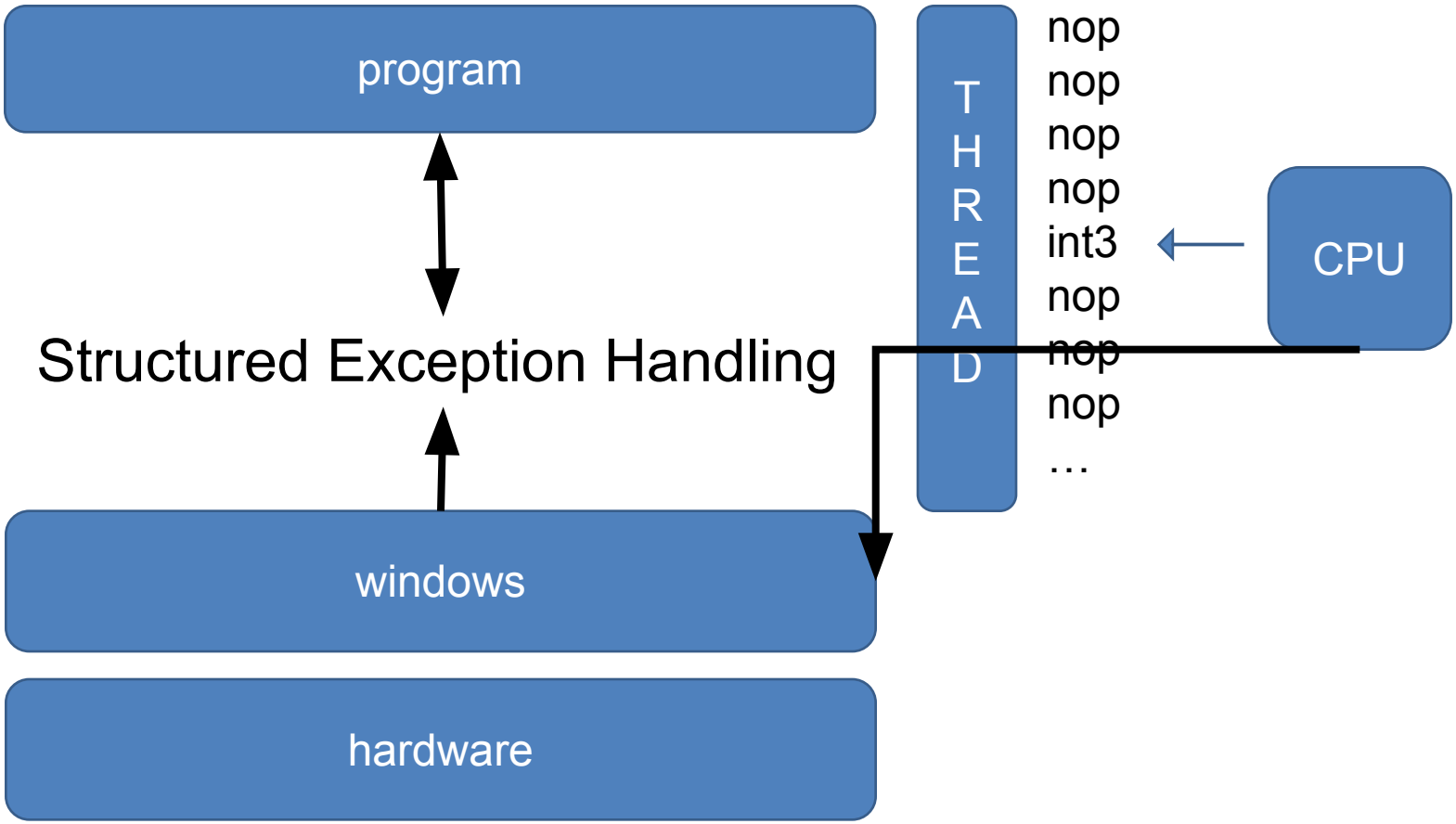
int 3


program
windows
hardware
THREAD
nop
nop
nop
nop
int3
nop
nop
nop
…
CPU
INTERRUPT

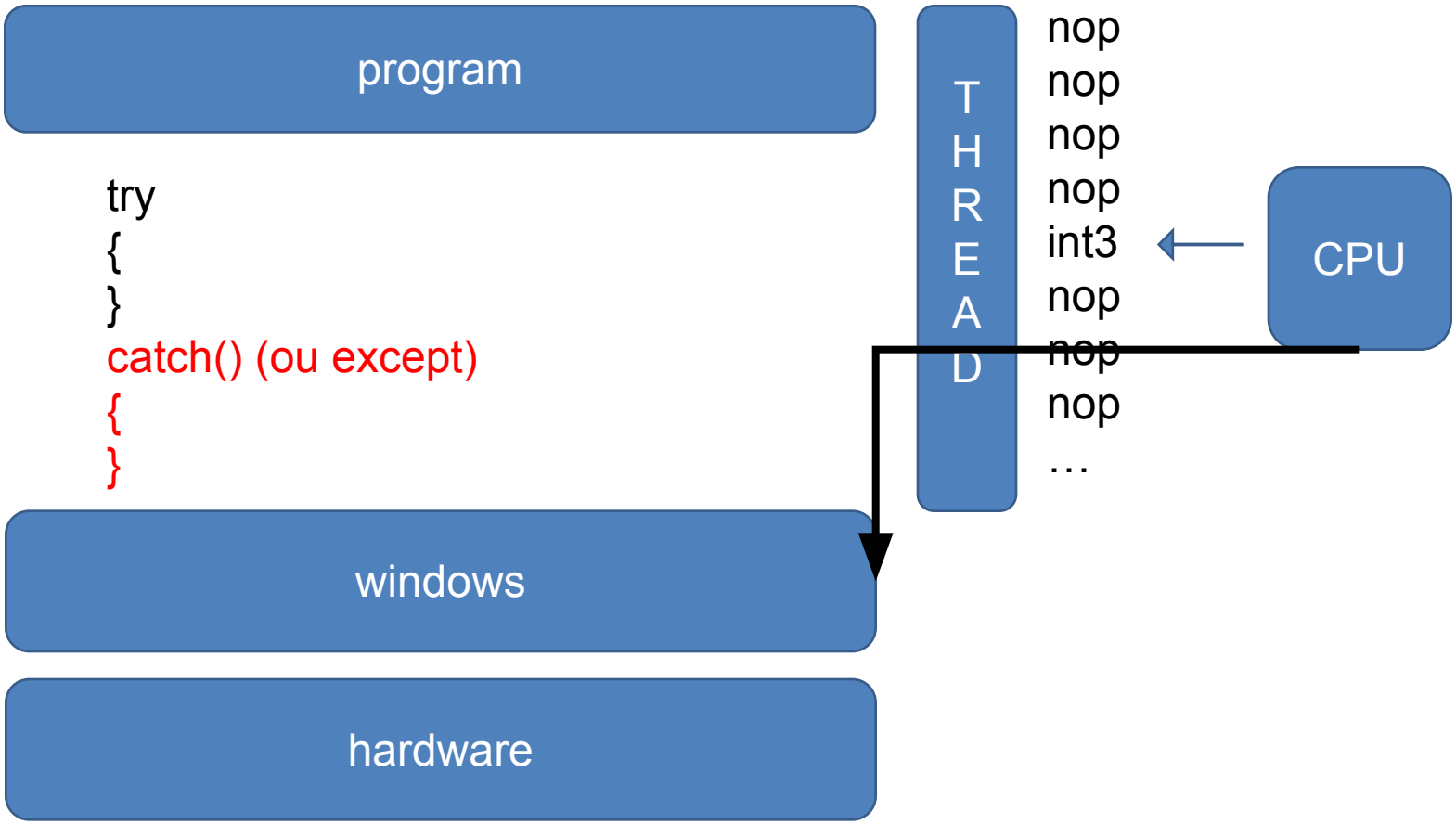
int 3


program
windows
hardware
THREAD
nop
nop
nop
nop
int3
nop
nop
nop
…
CPU

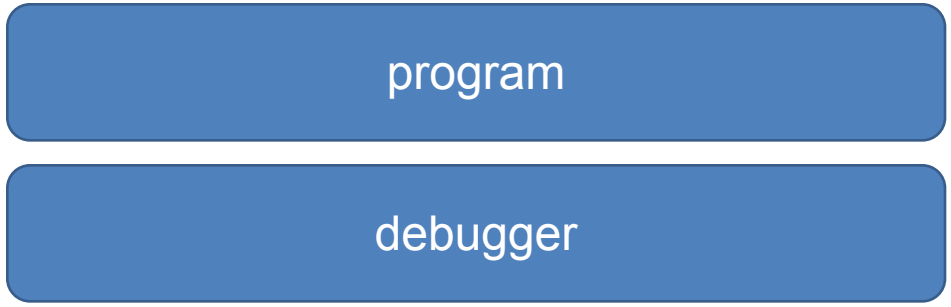
int 3


program
Structured Exception Handling
windows
hardware
THREAD
nop
nop
nop
nop
int3
nop
nop
nop
…
CPU

int 3


program
try
{
}
catch() (ou except)
{
}
windows
hardware
THREAD
nop
nop
nop
nop
int3
nop
nop
nop
…
CPU

int 3




```
try
{
}
catch() (ou except)
{
}
```

int 3


```
try
{
}
catch() (ou except)
{
}
```

int 3


```
try
{
}
catch() (ou except)
{
}
```

int 3

THE DEVELOPER'S CONFERENCE

program

program

```
try
{
}
catch() (ou except)
{
}
```

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

```
try
{
   // nonsense
   int 3 (DebugBreak())
}
except( ExceptFilter() )
{
   // nonsense
}

ExceptFilter()
{
   // here is the gold
}
```

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

```
try
{
   // nonsense
   int 3 (DebugBreak())
}
except( ExceptFilter() )
{
   // nonsense
}

ExceptFilter()
{
   // here is the gold
}
```

Globalcode

int 3

THE DEVELOPER'S CONFERENCE

```
try
{
   // nonsense
   int 3 (DebugBreak())
}
except( ExceptFilter() )
{
   // nonsense
}

ExceptFilter()
{
   // here is the gold
}
```

Globalcode

int 3

“Run, code, run!” – No One

int 3

THE DEVELOPER'S CONFERENCE

- Problemas:
  - Multithreading (e lock, e mutex, e inferno).
    - Fluxo não-contínuo de execução
    - Performance
    - Fica feio

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

Long Jump Silver!

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

Code
Code
Code
Code
SetLongJump

Code
Code
Code
…
Jump!

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

Code
Code
Code
Code
SetLongJump
Code
Code
Code
…
Jump!

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

Code
Code
Code
Code
SetLongJump
Code
Code
Code
…
Jump!

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

Code
Code
Code
Code
SetLongJump
Code
Code
Code
…
Jump!

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

Code
Code
Code
Code
SetLongJump
Code
Code
Code
…
Jump!

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

Code
Code
Code
Code
SetLongJump
Code
Code
Code
…
Jump!

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

```
#define ANTIDEBUG(code)
{
     jmp_buf env;

     if( setjmp(env) == 0 )
     {
          LongJmp(&env);
     }
     else
     {
          code;
     }
}
```

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

```
#define ANTIDEBUG(code)
{
     jmp_buf env;

     if( setjmp(env) == 0 )
     {
          LongJmp(&env);
     }
     else
     {
          code;
     }
}
```

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

```
DWORD LongJmp(jmp_buf* env)
{
    __try
    {
        __asm int 3
    }
        __except( EXCEPTION_EXECUTE_HANDLER )
    {
        longjmp(*env, 1);
    }

    return ERROR_SUCCESS;
}
```

Globalcode

int 3: v. 2

THE DEVELOPER'S CONFERENCE

```
DWORD LongJmp(jmp_buf* env)
{
    __try
    {
        __asm int 3
    }
        __except( EXCEPTION_EXECUTE_HANDLER )
    {
        longjmp(*env, 1);
    }

    return ERROR_SUCCESS;
}
```

Globalcode

int 3: v. 2

“Run, Forrest, run!” – Long Dong

Debug Port

THE DEVELOPER'S CONFERENCE

Globalcode

Debug Port

THE DEVELOPER'S CONFERENCE

Globalcode

Debug Port

THE DEVELOPER'S CONFERENCE

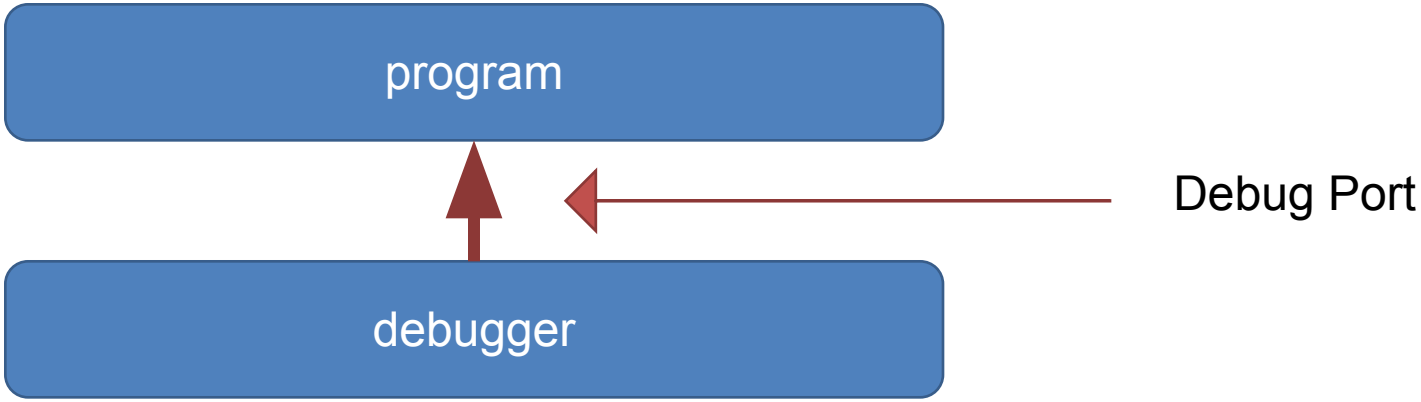
program

debugger

```
try
{
}
catch() (ou except)
{
}
```

Globalcode

Debug Port




```
try
{
}
catch() (ou except)
{
}
```

Debug Port

THE DEVELOPER'S CONFERENCE

Como é o código de um depurador:

Globalcode

Debug Port

THE DEVELOPER'S CONFERENCE

Como é o código de um depurador:

```
Loop:
  WaitForDebugEvent(&debugEvt, INFINITE);
  ContinueDebugEvent(pid, tid, DBG_SBRUBLES);
```

Globalcode

Debug Port

THE DEVELOPER'S CONFERENCE

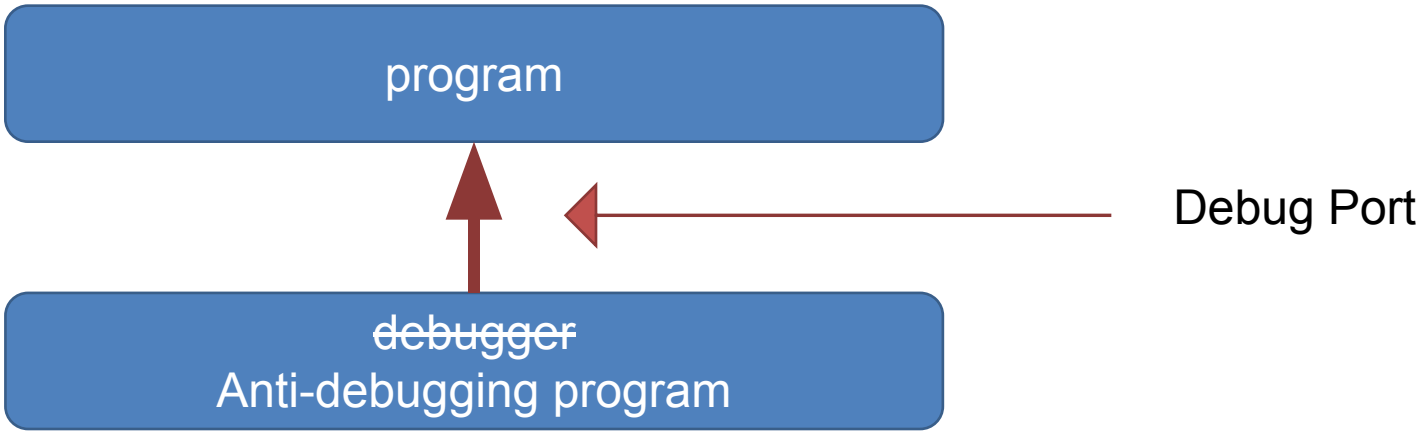
Como é o código de um depurador:

```
Loop:
  WaitForDebugEvent(&debugEvt, INFINITE);
  ContinueDebugEvent(pid, tid, DBG_SBRUBLES);
```

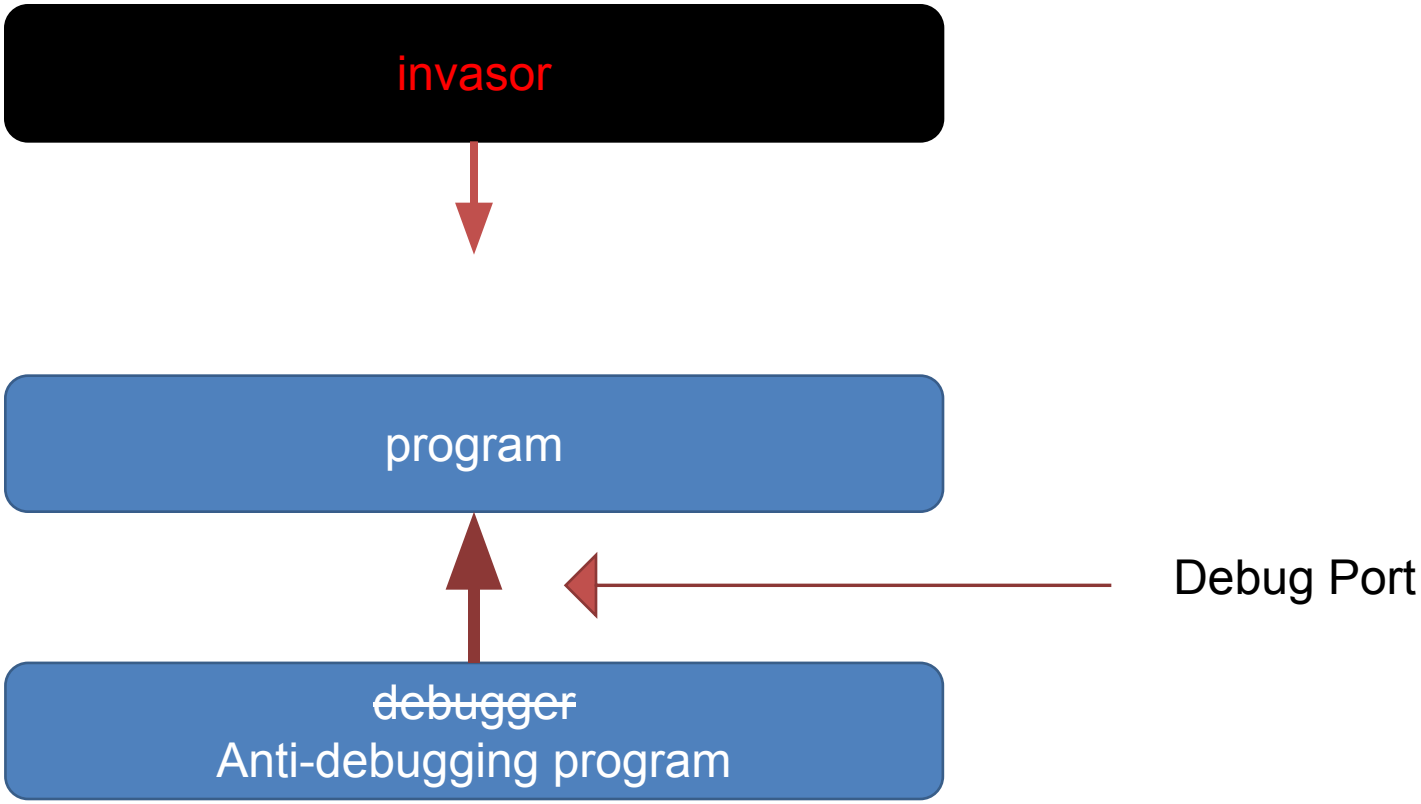
That's it!

Globalcode

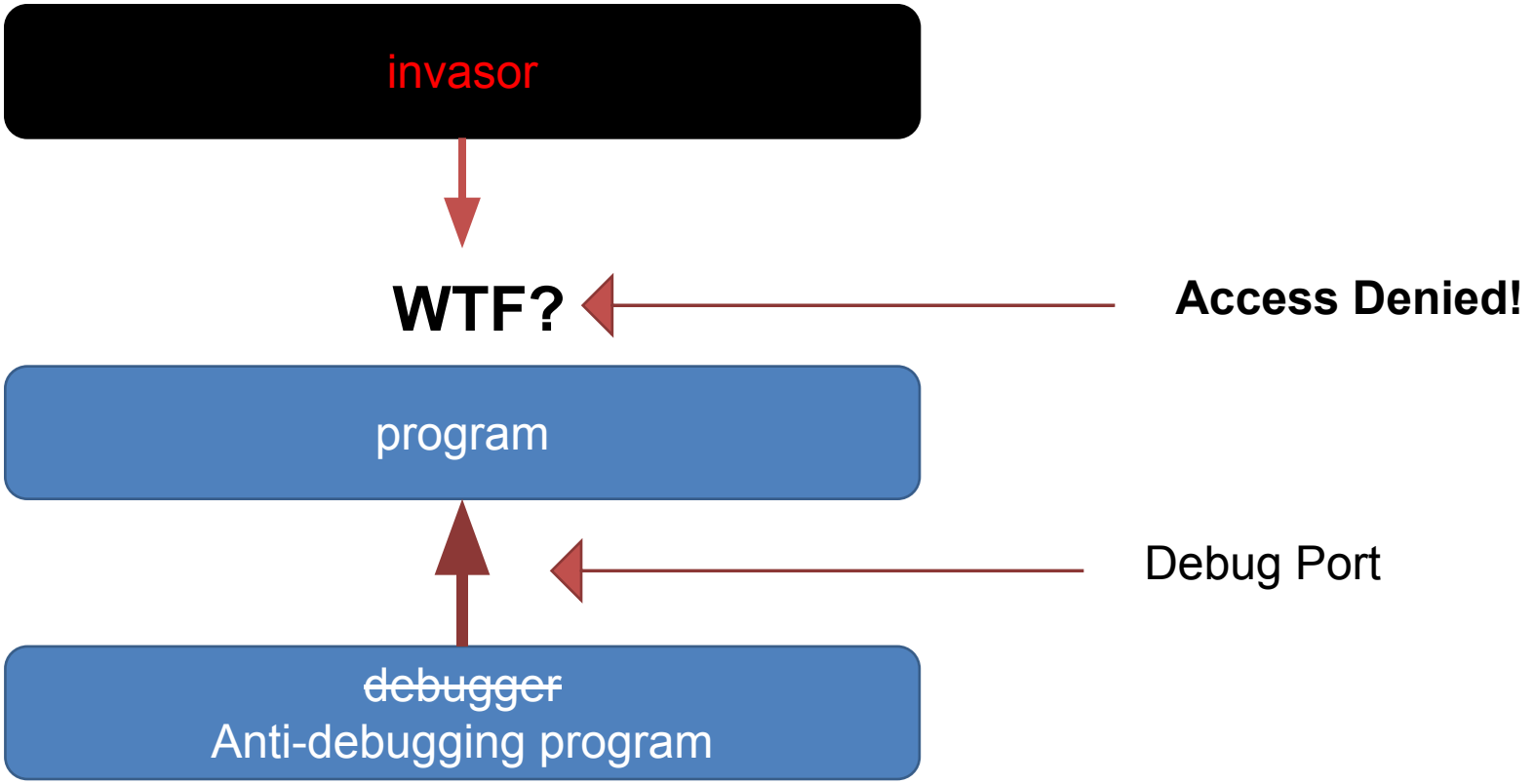
Debug Port



Debug Port


invasor
program
Debug Port
~~debugger~~
Anti-debugging program

Debug Port


invasor
WTF?
Access Denied!
program
Debug Port
~~debugger~~
Anti-debugging program

Debug Port

“Knock
Knock
Knockin'
on debug's port”

Debug Port

“Knock
Knock
Knockin'
on debug's port”

- Bob Dybug

Attach

THE DEVELOPER'S CONFERENCE

Did you say…

Globalcode

Attach

THE DEVELOPER'S CONFERENCE

assembly

??????????

Globalcode

Attach

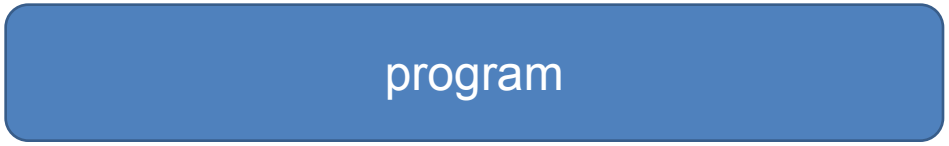
THE DEVELOPER'S CONFERENCE

```
// opcodes to run a jump to
// the function AntiAttachAbort

BYTE jmpToAntiAttachAbort[] =
{
   0xB8, 0xCC, 0xCC, 0xCC, 0xCC,
   // mov eax, 0xCCCCCCCC

   0xFF, 0xE0
   // jmp eax

};
```

Globalcode

Attach

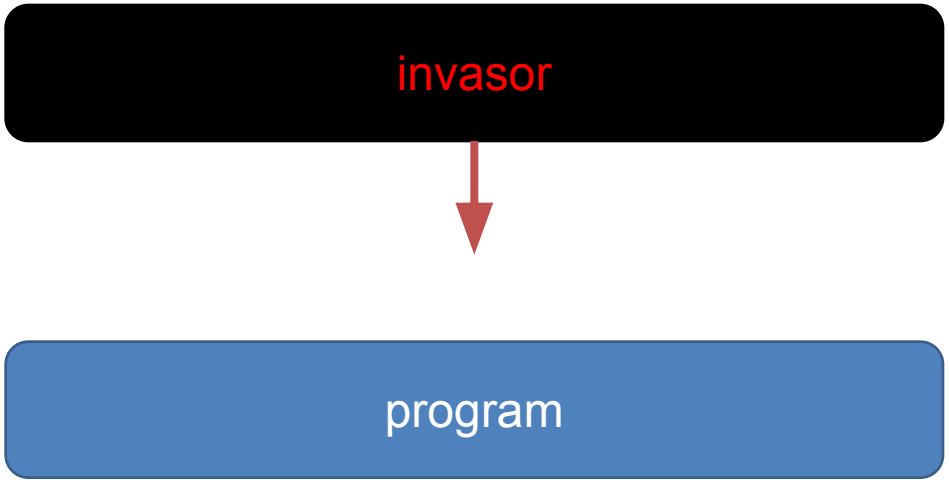
THE DEVELOPER'S CONFERENCE



Globalcode

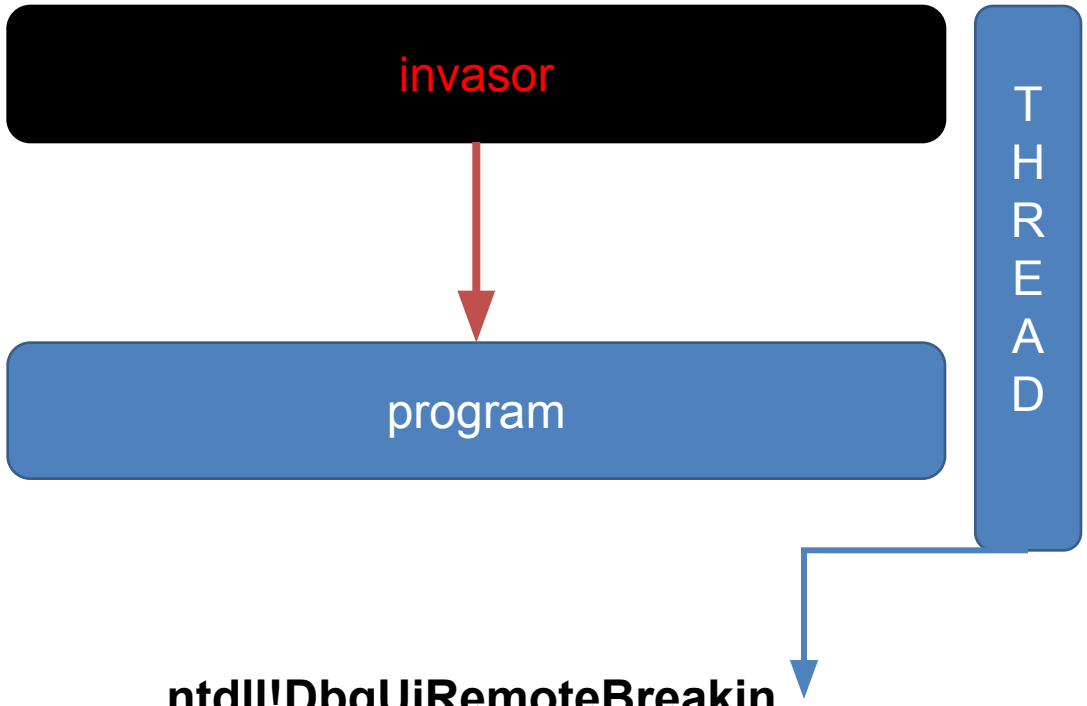
Attach

THE DEVELOPER'S CONFERENCE



Globalcode

Attach



Attach

THE DEVELOPER'S CONFERENCE

ntdll!DbgUiRemoteBreakin

```
773F10A0  push        8
773F10A2  push        773F10F8h
773F10A7  call        __SEH_prolog4 (77384420h)
773F10DB  xor         eax,eax
773F10DD  inc         eax
773F10DE  ret
773F10DF  mov         esp,dword ptr [ebp-18h]
773F10E2  mov         dword ptr [ebp-4],0FFFFFFFEh
773F10E9  push        0
773F10EB  call        RtlExitUserThread (77362B10h)
773F10F0  int         3
```

Globalcode

Attach

THE DEVELOPER'S CONFERENCE

ntdll!DbgUiRemoteBreakin

```
773F10A0  push        8
773F10A2  push        773F10F8h
773F10A7  call        __SEH_prolog4 (77384420h)
773F10DB  xor         eax,eax
773F10DD  inc         eax
773F10DE  ret
773F10DF  mov         esp,dword ptr [ebp-18h]
773F10E2  mov         dword ptr [ebp-4],0FFFFFFFEh
773F10E9  push        0
773F10EB  call        RtlExitUserThread (77362B10h)
773F10F0  int         3
```

Globalcode

Attach

THE DEVELOPER'S CONFERENCE

ntdll!DbgUiRemoteBreakin

```
773F10A0  push        8
773F10A2  push        773F10F8h
773F10A7  call        __SEH_prolog4 (77384420h)
773F10DB  xor         eax,eax
773F10DD  inc         eax
773F10DE  ret
773F10DF  mov         esp,dword ptr [ebp-18h]
773F10E2  mov         dword ptr [ebp-4],0FFFFFFFEh
773F10E9  push        0
773F10EB  call        RtlExitUserThread (77362B10h)
773F10F0  int         3
```

Globalcode

Attach

THE DEVELOPER'S CONFERENCE

ntdll!DbgUiRemoteBreakin

```
773F10A0  jmp         NaNaNiNaNaaaaooooo

773F10A7  call        __SEH_prolog4 (77384420h)
773F10DB  xor         eax,eax
773F10DD  inc         eax
773F10DE  ret
773F10DF  mov         esp,dword ptr [ebp-18h]
773F10E2  mov         dword ptr [ebp-4],0FFFFFFFEh
773F10E9  push        0
773F10EB  call        RtlExitUserThread (77362B10h)
773F10F0  int         3
```

Globalcode

Attach

THE DEVELOPER'S CONFERENCE

ntdll!DbgUiRemoteBreakin

```
773F10A0  jmp         AntiAttachAbort

773F10A7  call        __SEH_prolog4 (77384420h)
773F10DB  xor         eax,eax
773F10DD  inc         eax
773F10DE  ret
773F10DF  mov         esp,dword ptr [ebp-18h]
773F10E2  mov         dword ptr [ebp-4],0FFFFFFFEh
773F10E9  push        0
773F10EB  call        RtlExitUserThread (77362B10h)
773F10F0  int         3
```

Globalcode

Attach

THE DEVELOPER'S CONFERENCE

AntiAttachAbort?

Globalcode

Attach

THE DEVELOPER'S CONFERENCE

AntiAttachAbort?

Globalcode

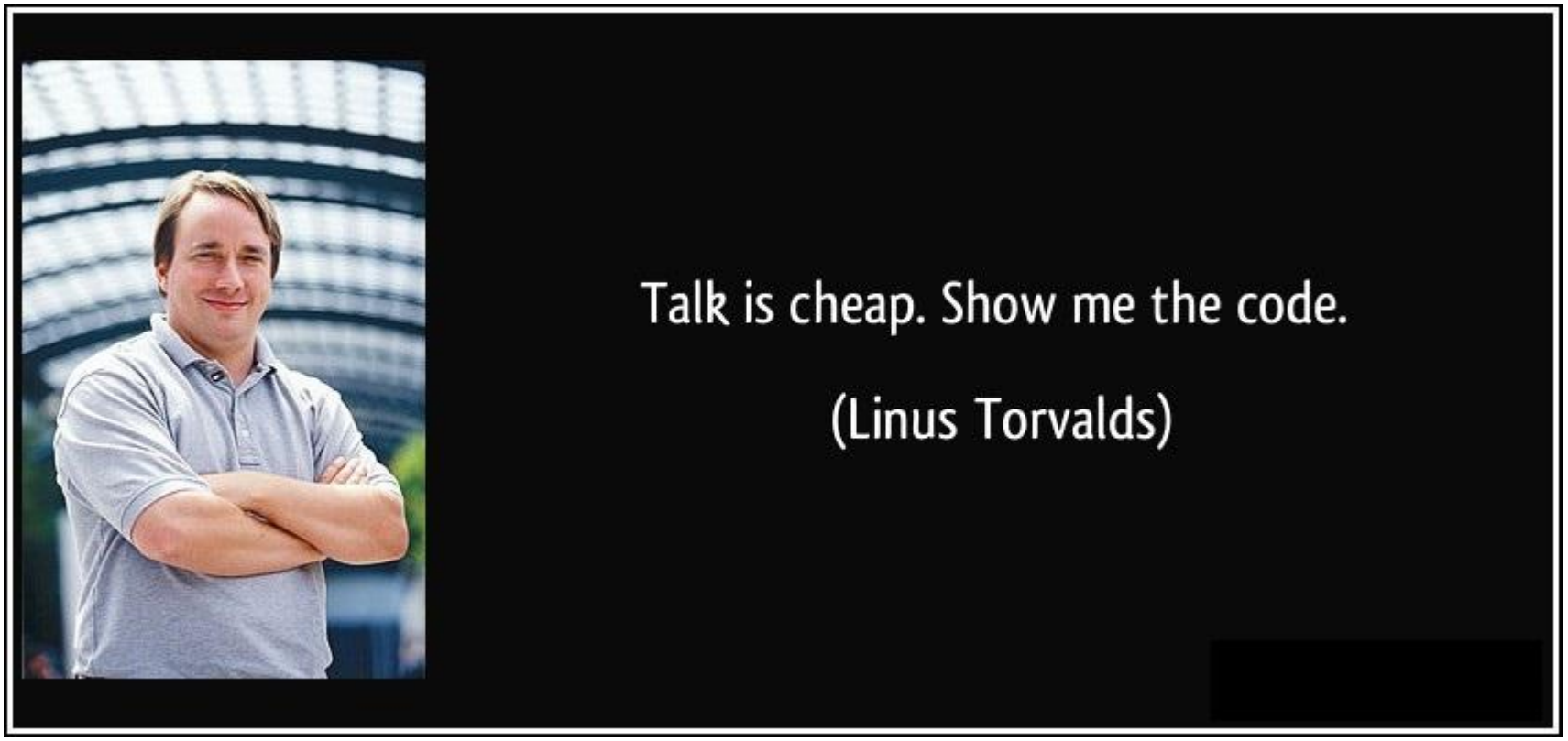
Attach

THE DEVELOPER'S CONFERENCE

AntiAttachAbort?

Globalcode

Attach



Conclusão

THE
DEVELOPER'S
CONFERENCE

Globalcode

Conclusão

THE DEVELOPER'S CONFERENCE

Globalcode

Conclusão

THE DEVELOPER'S CONFERENCE

- Técnicas anti-debugging são complicadas
  - TODO: Encapsular em uma LIB
- Nenhuma técnica é perfeita
  - Performance, complexidade, instabilidade…
- Linus Torvalds pode aparecer em um slide de um MVP e ele não será expulso da congregação
  - O contrário não é verdadeiro

Globalcode

Contato

THE DEVELOPER'S CONFERENCE

e-mail

wanderley@caloni.com.br

twitter

site

Globalcode

Agradecimentos

THE DEVELOPER'S CONFERENCE

Globalcode