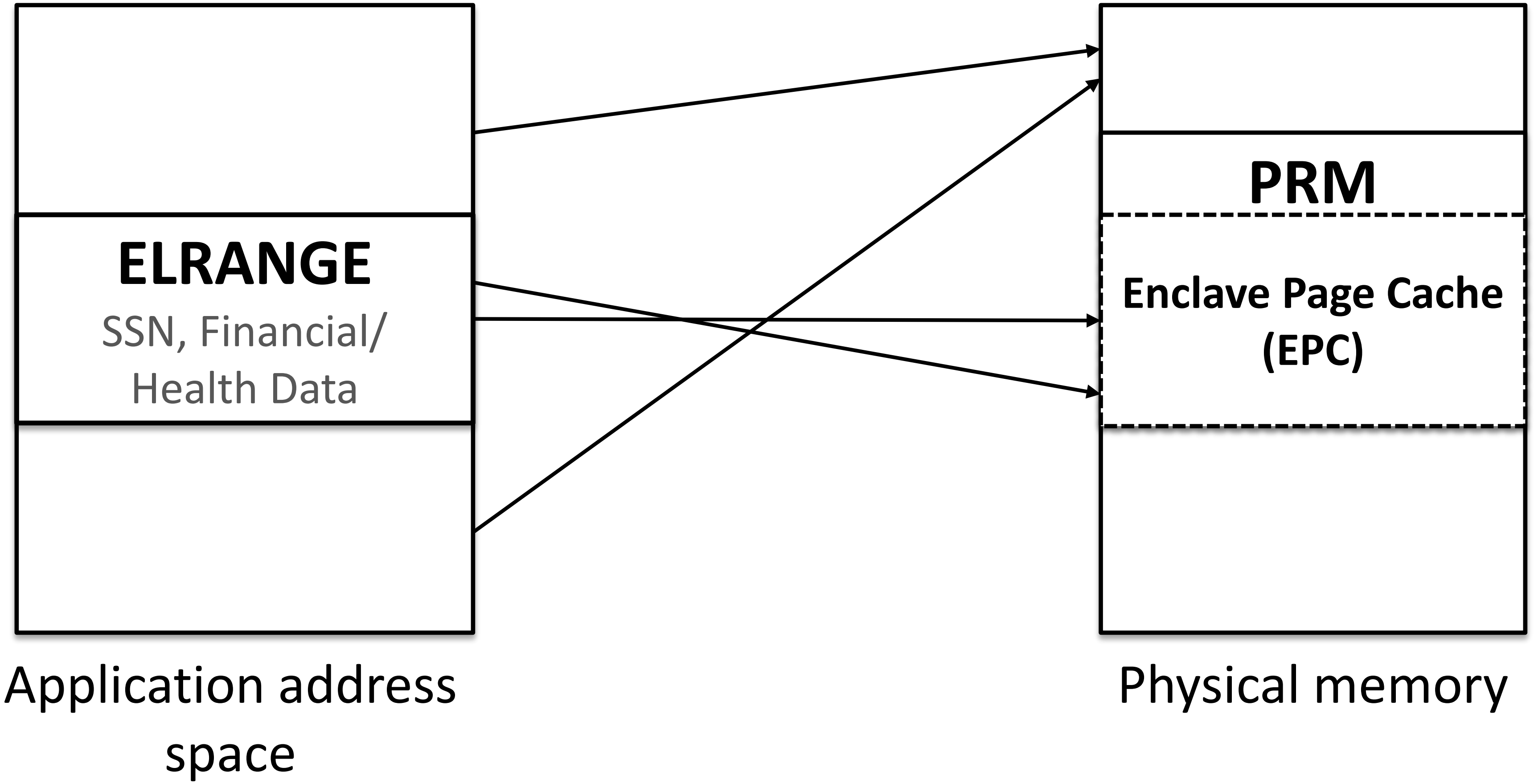
Leaky Cauldron on the Dark Land: Understanding Memory Side-Channel Hazards in SGX

[1,4]Wenhao Wang, [2]Guoxing Chen, [1]Xiaorui Pan, [2]Yinqian Zhang, [1]XiaoFeng Wang,
[3]Vincent Bindschaedler, [1]Haixu Tang and [3]Carl A. Gunter

[1]Indiana University Bloomington [2]The Ohio State University
[3]University of Illinois Urbana-Champaign [4]Institute of Information Engineering

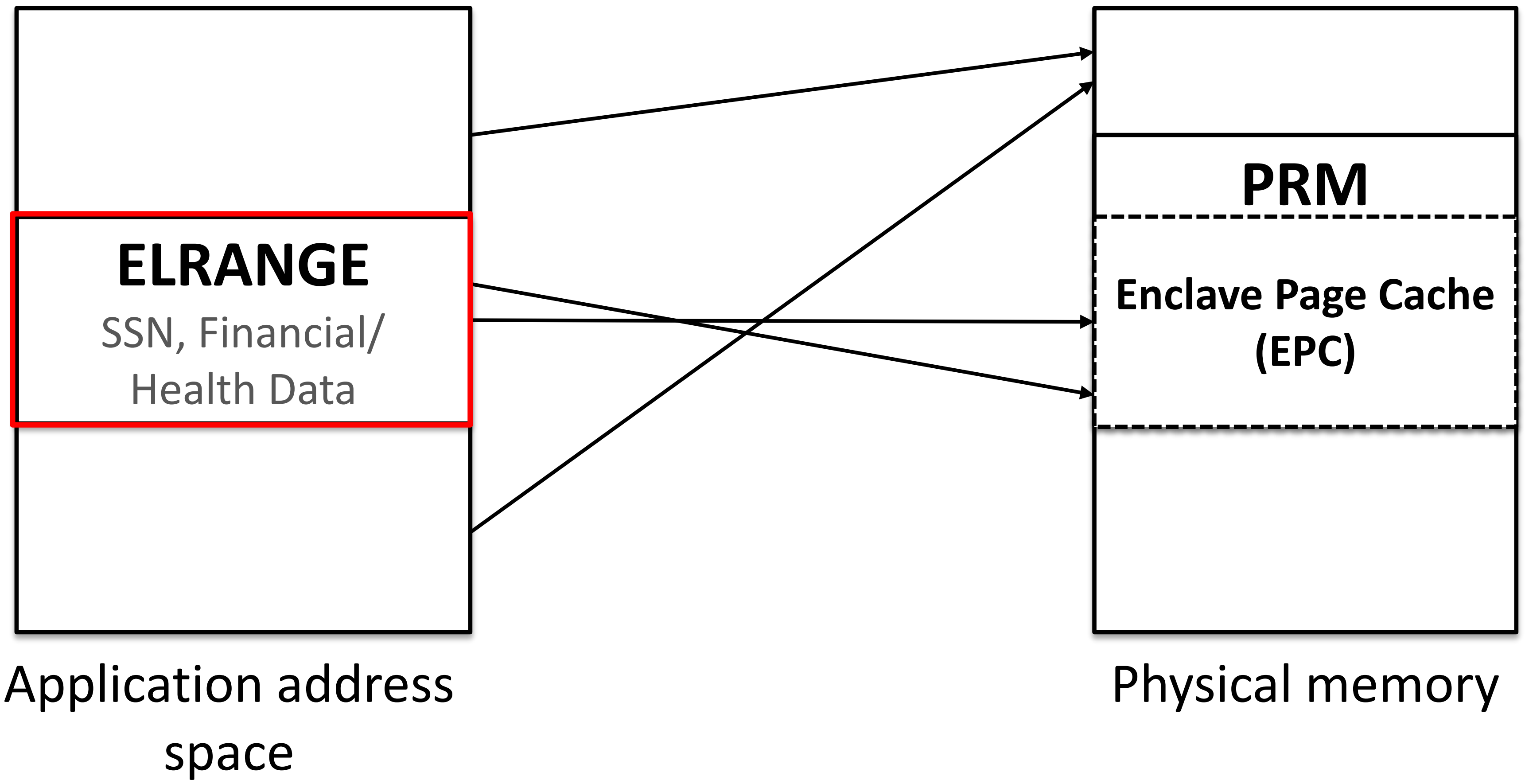
Intel Software Guard Extensions

Processor Reserved Memory (PRM)



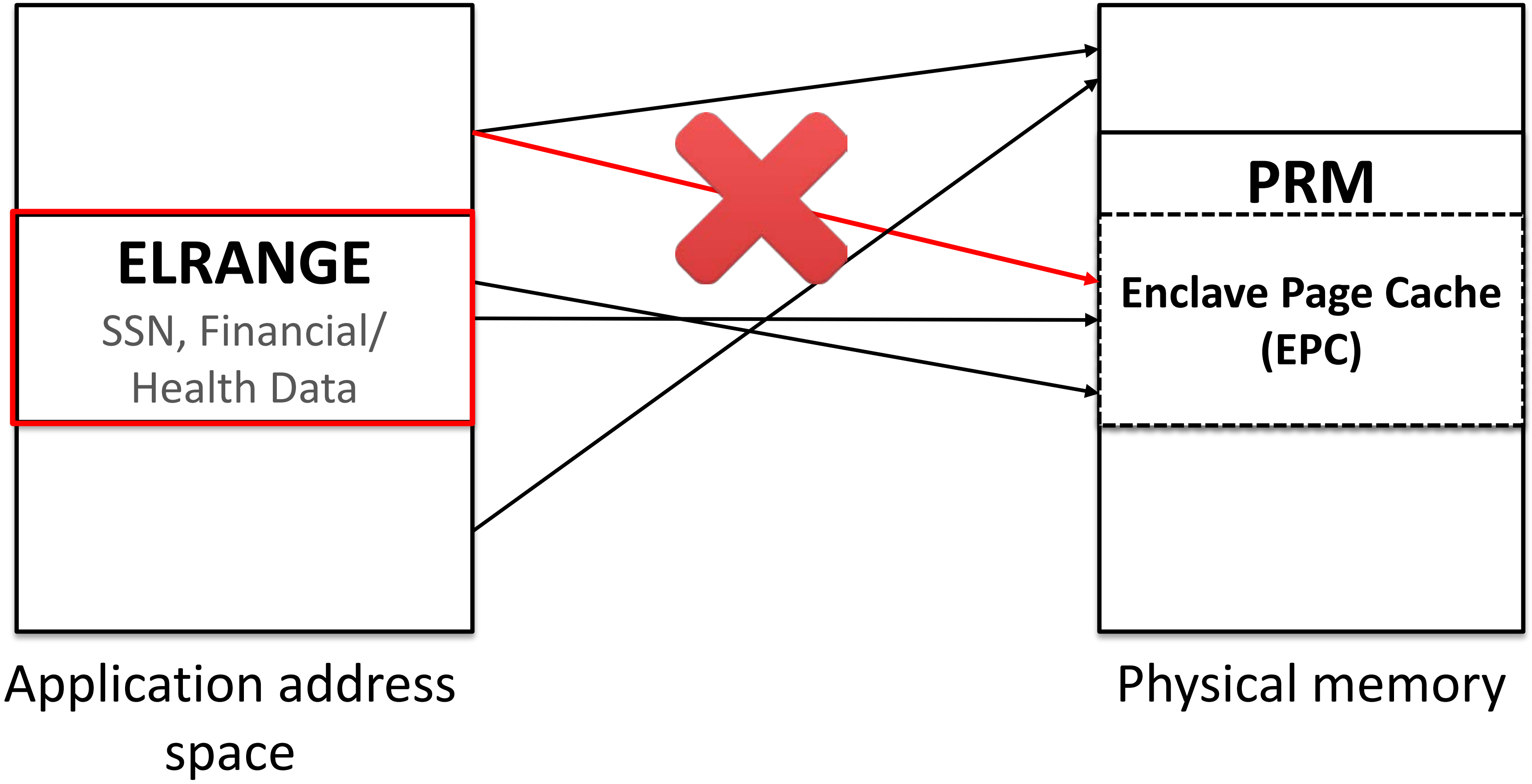
Intel Software Guard Extensions

Processor Reserved Memory (PRM)



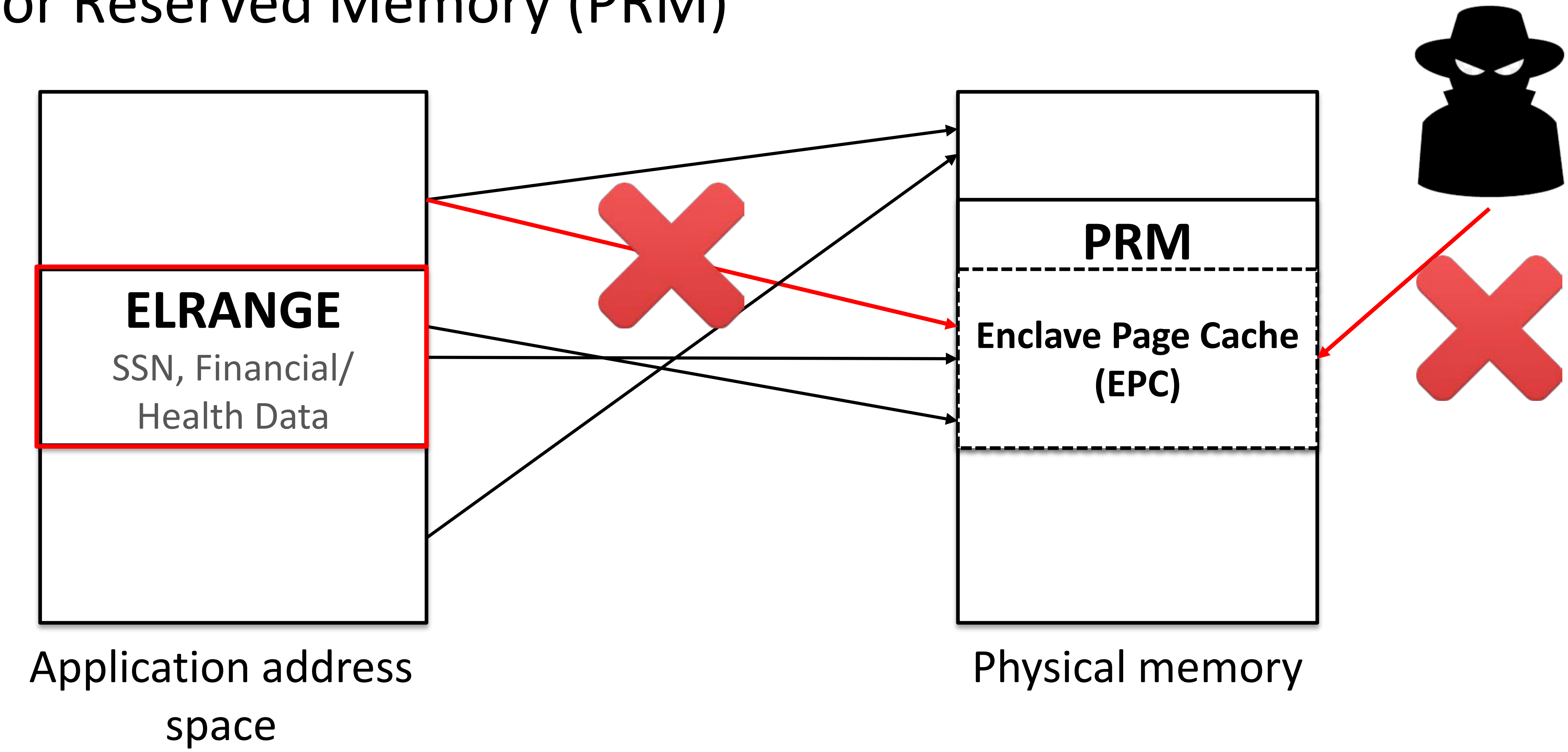
Intel Software Guard Extensions

Processor Reserved Memory (PRM)



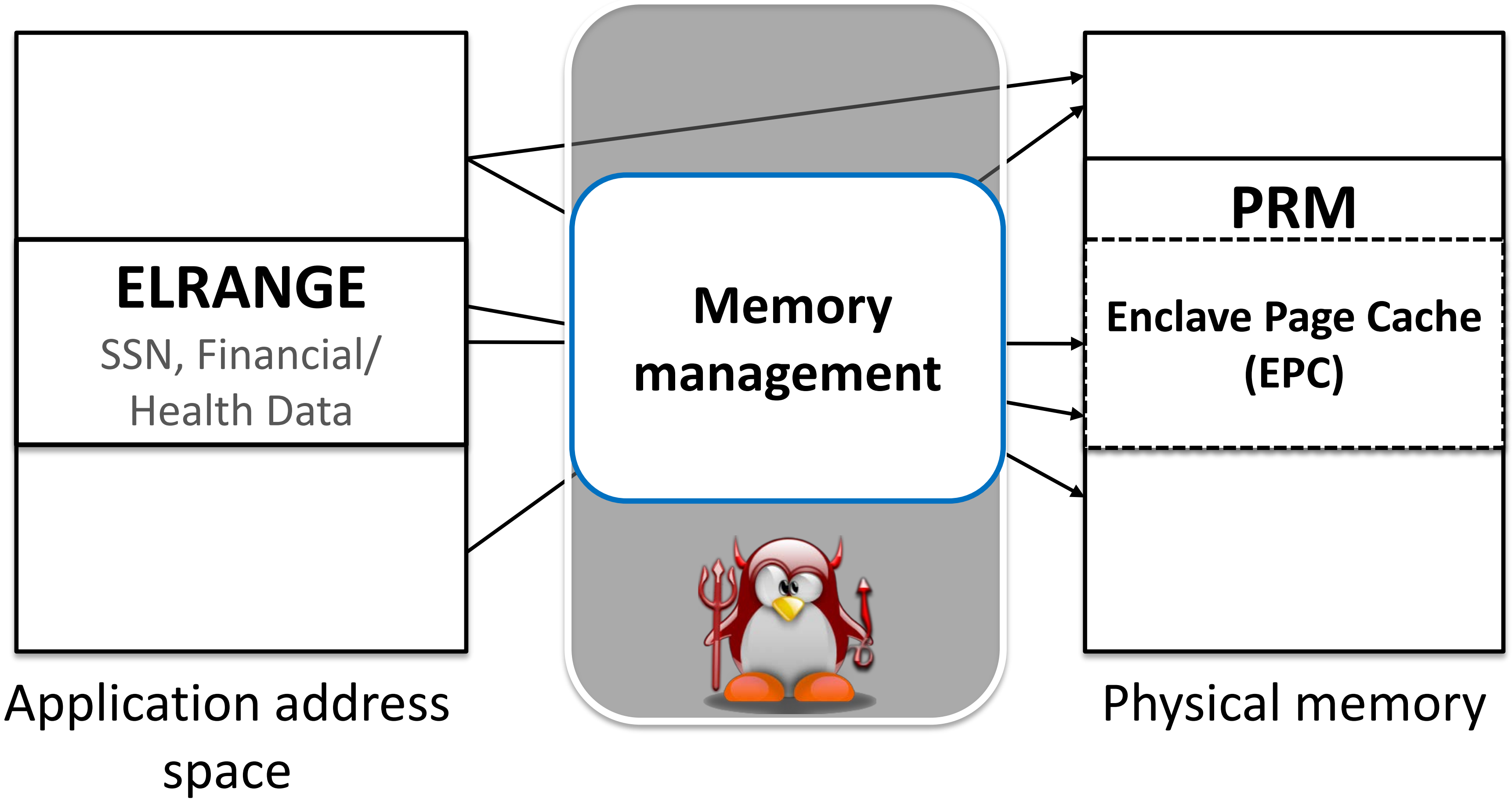
Intel Software Guard Extensions

Processor Reserved Memory (PRM)



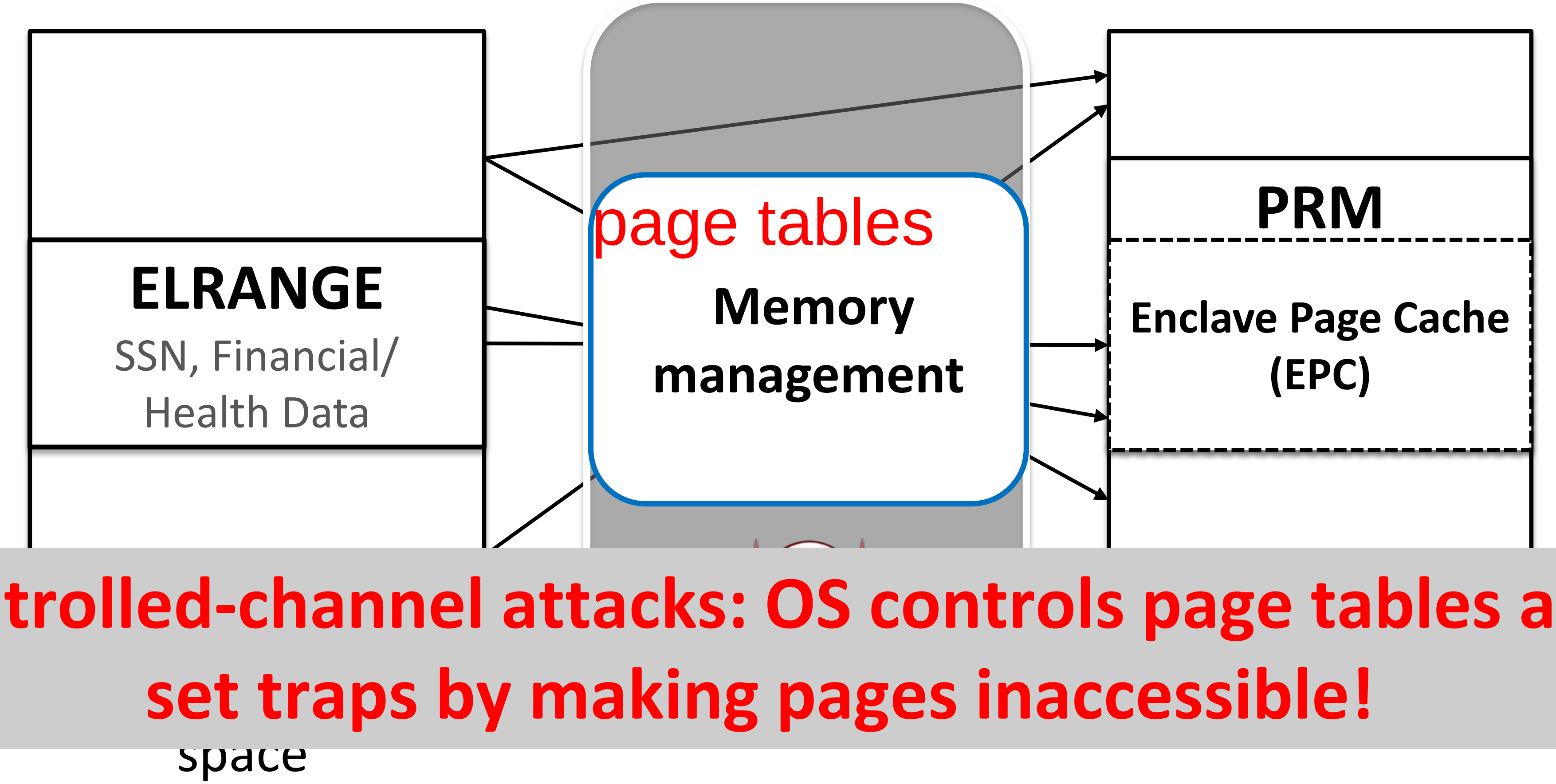
Intel Software Guard Extensions

Processor Reserved Memory (PRM)



Intel Software Guard Extensions

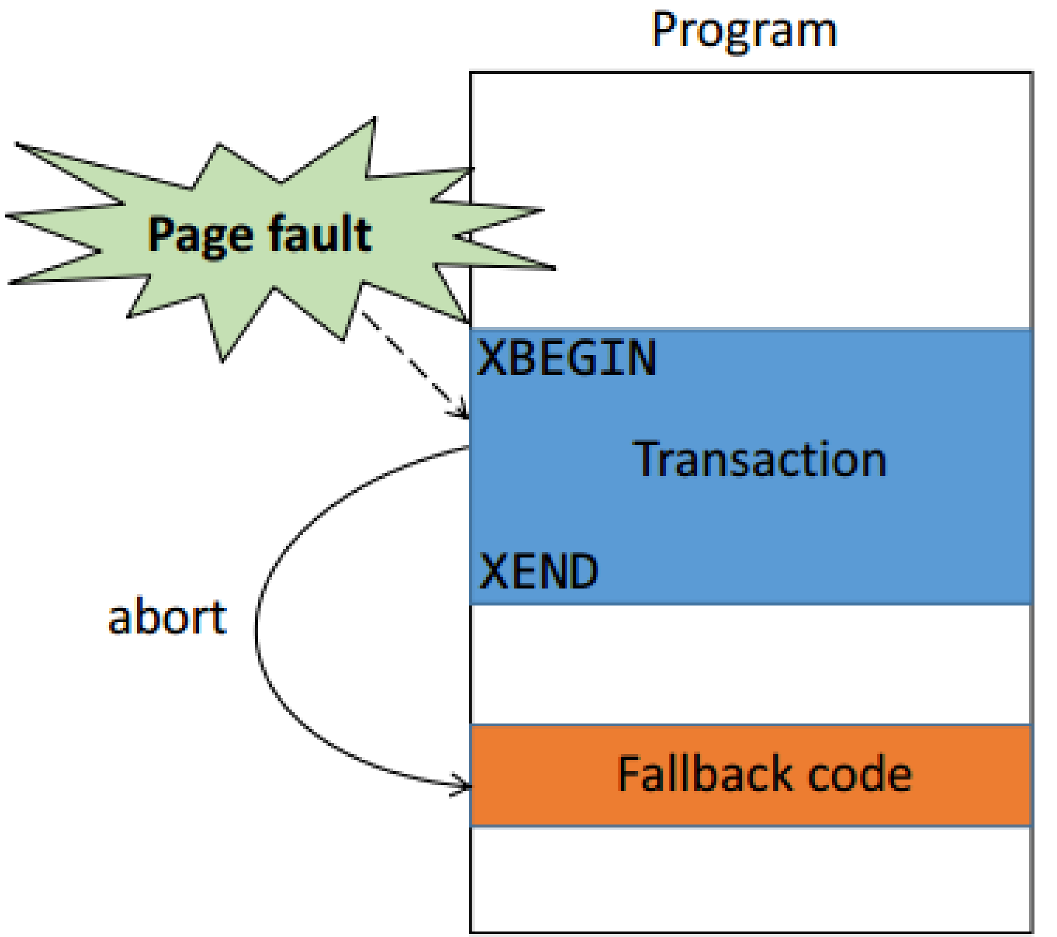
Processor Reserved Memory (PRM)



Controlled-channel attacks: OS controls page tables and set traps by making pages inaccessible!

Defenses against page-fault attacks

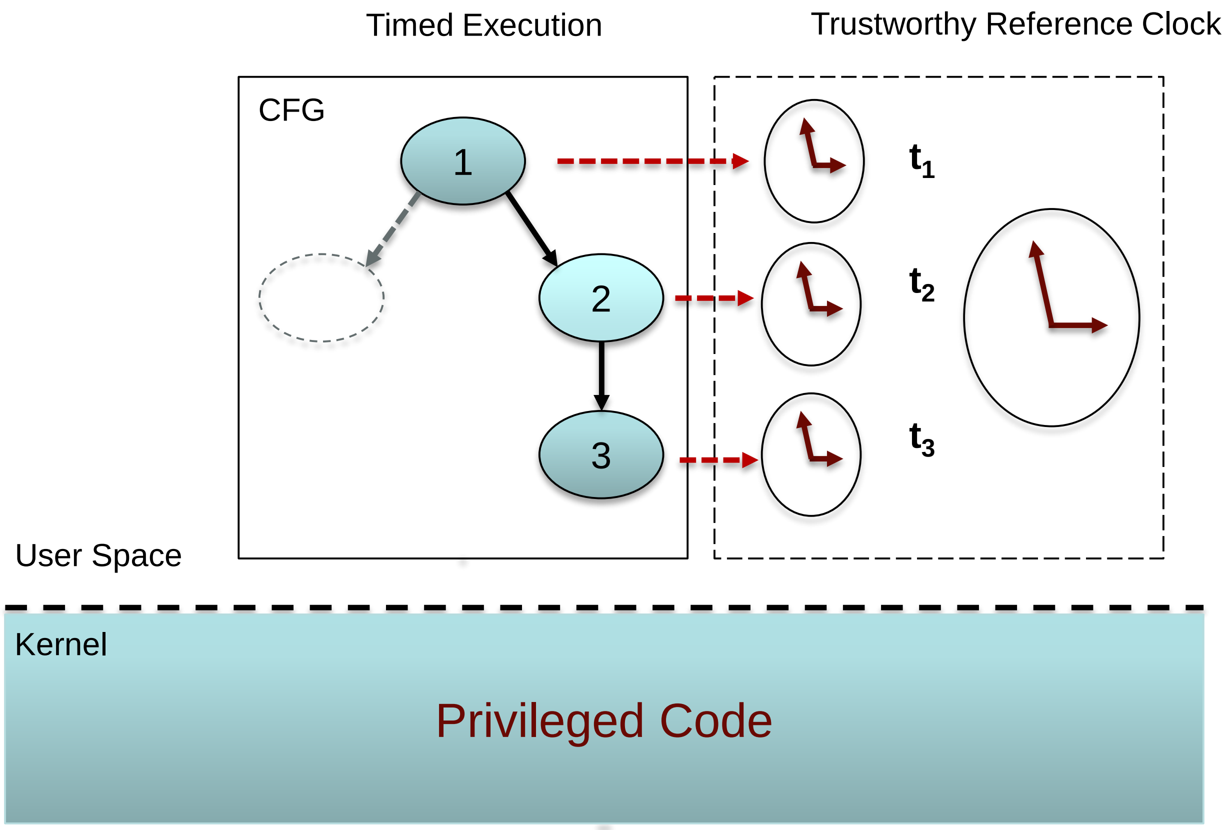
T-SGX



Images taken from the authors' slides

Defenses against page-fault attacks

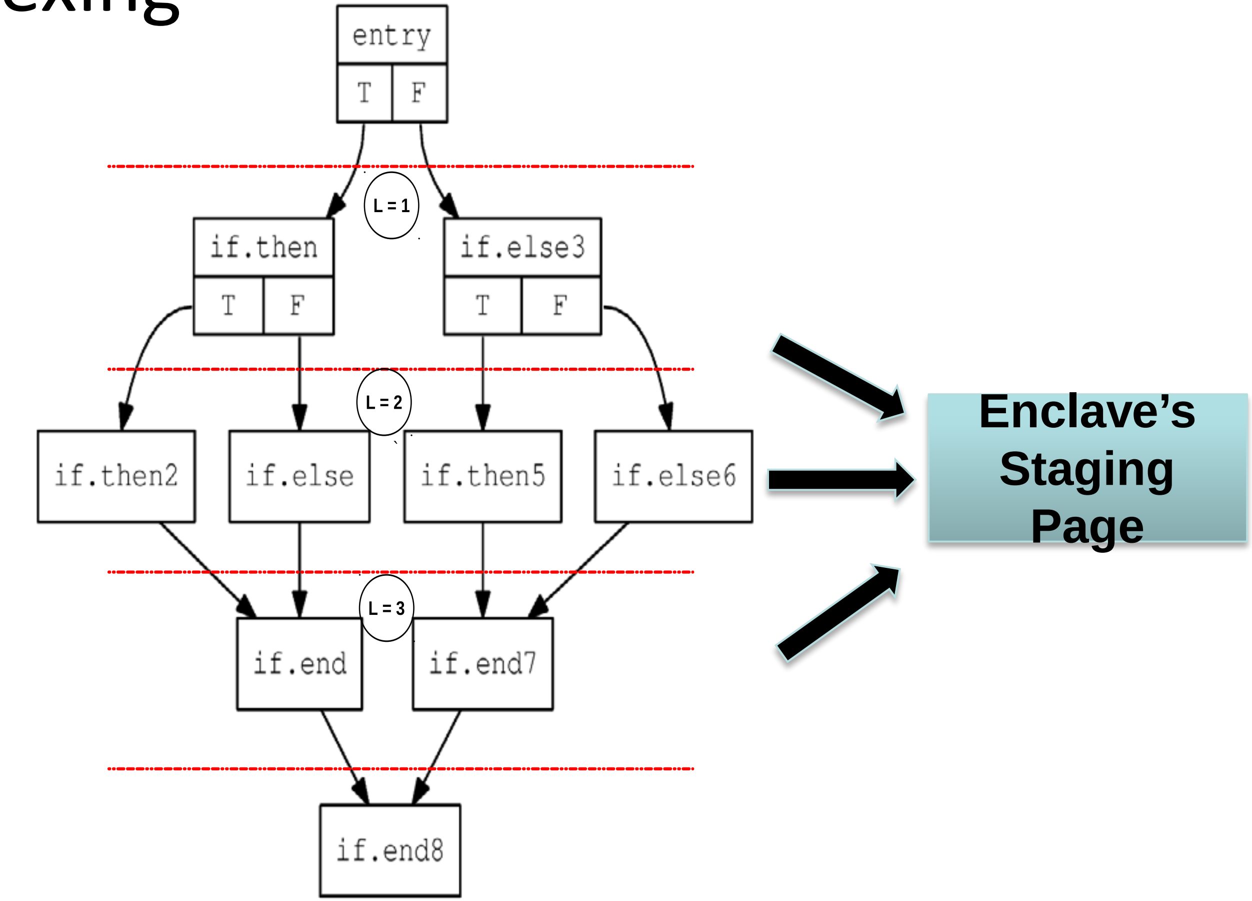
DEJA VU



Images taken from the authors' slides

Defenses against page-fault attacks

Deterministic multiplexing

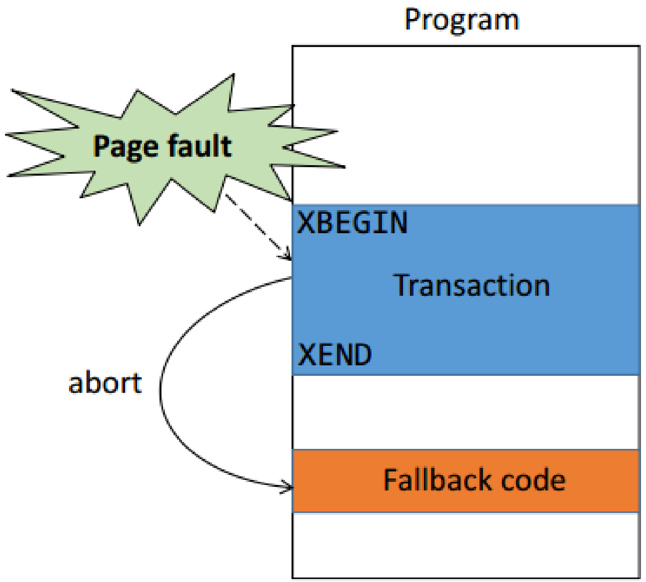


Enclave’s Staging Page

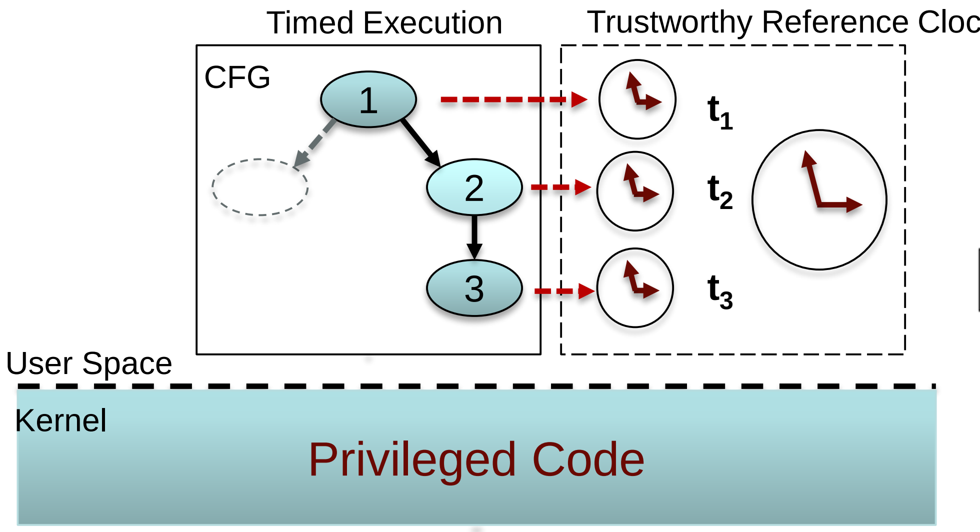
Images taken from the authors’ slides

Defenses against page-fault attacks

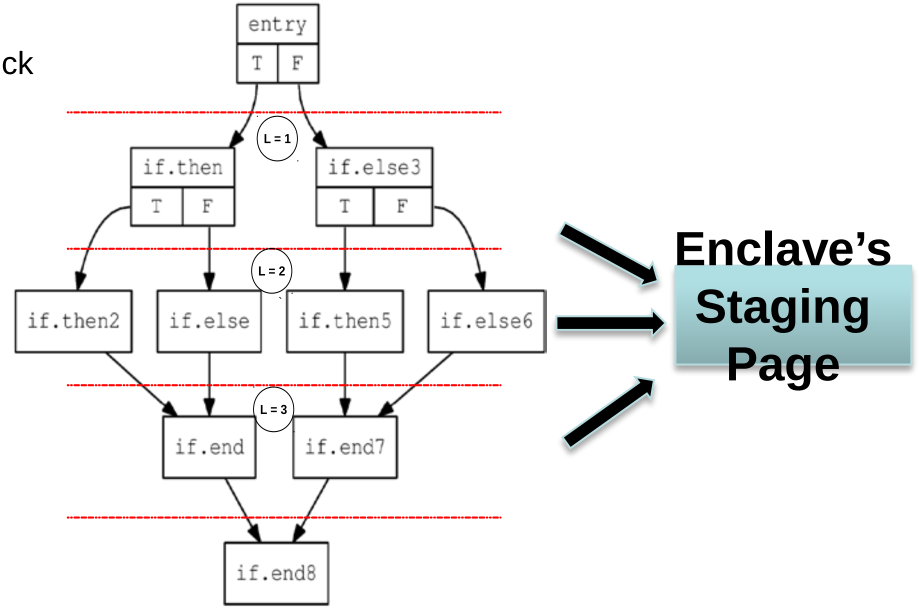
T-SGX



DEJA VU



Deterministic multiplexing



Images taken from the authors' slides

Our contributions

- A comprehensive understanding of SGX memory side channels.
  - 8 attack vectors.

Our contributions

- A comprehensive understanding of SGX memory side channels.
  - 8 attack vectors.

- Reducing AEXs induced by page level attacks.
  - A new type of attacks.

Our contributions

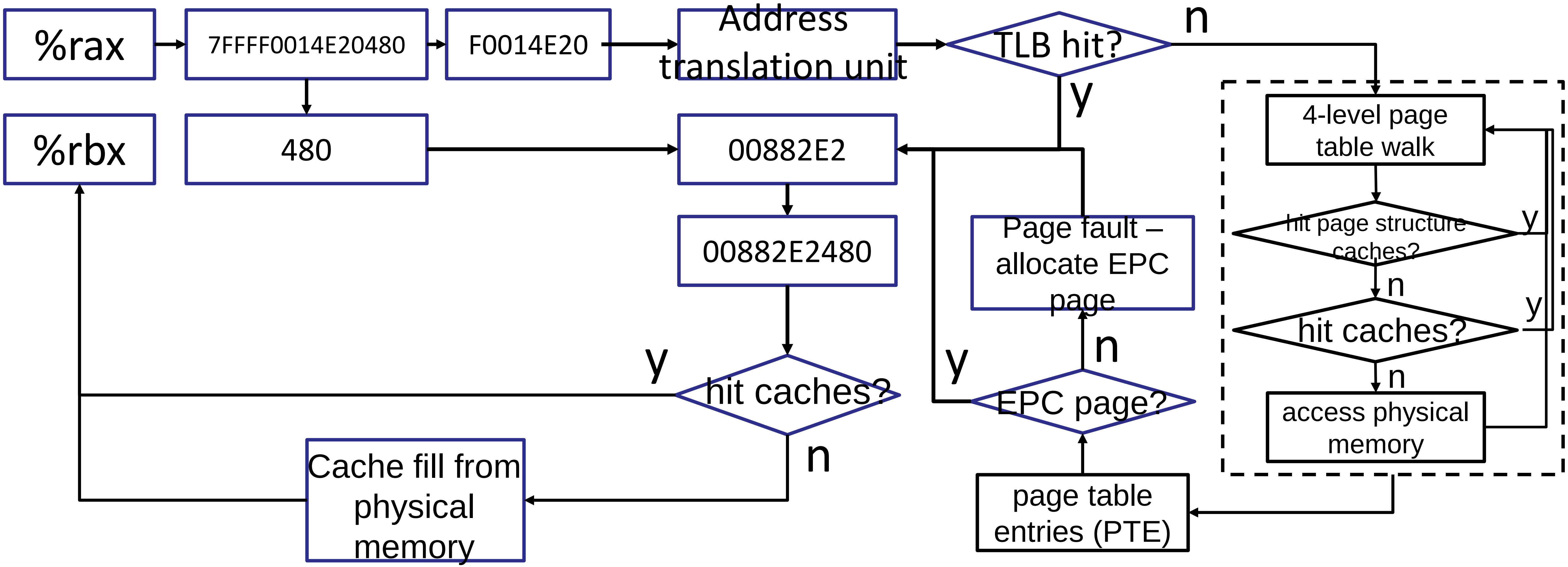
- A comprehensive understanding of SGX memory side channels.
  - 8 attack vectors.

- Reducing AEXs induced by page level attacks.
  - A new type of attacks.

- Achieving finer-grained (than 4 KB) spatial granularity.
  - Cache-DRAM attack.

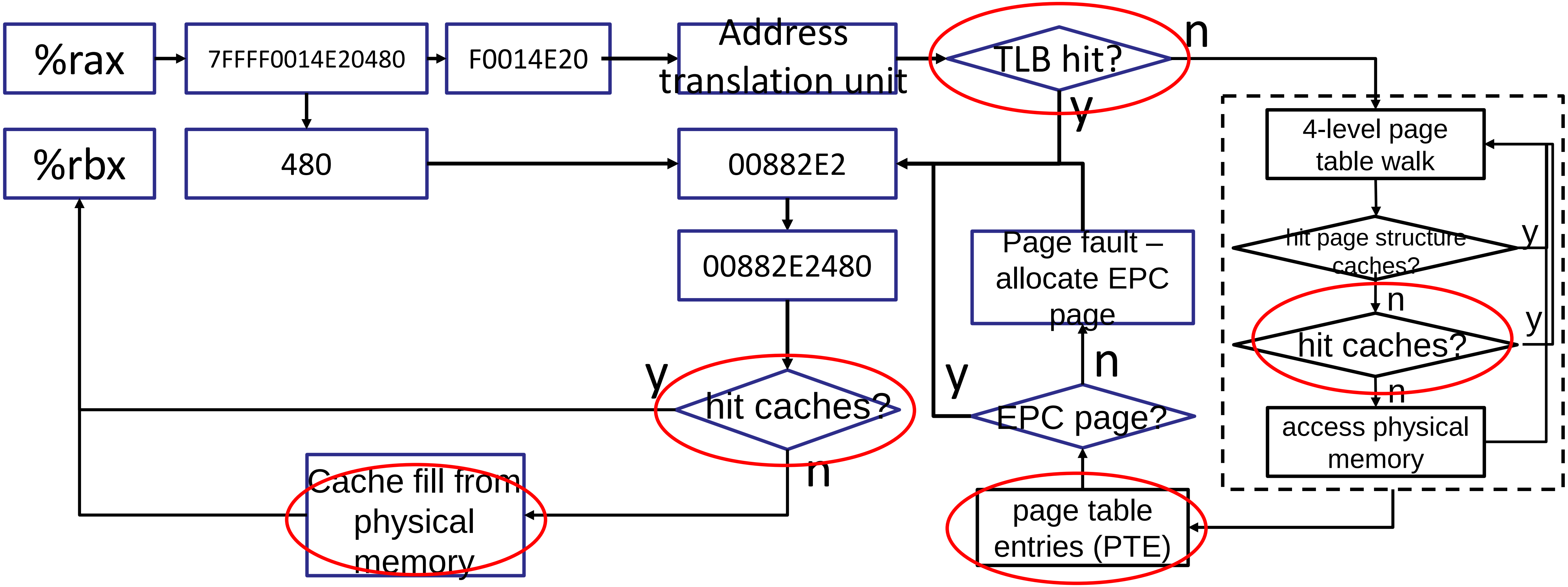
1. Understanding Attack Surfaces

mov (%rax), %rbx



1. Understanding Attack Surfaces

mov (%rax), %rbx



Summary of Attack vectors

- □ V1. Shared TLB entries under HT.
- □ V2. Selective TLB entries flushing without HT.
- □ V3. Referenced PTEs are cached as data.
- □ V4. Updates of accessed flags.
- □ V5. Updates of dirty flags.
- □ V6. Triggering page faults with P/X or reserved bits.
- □ V7. CPU caches are shared between the enclave and non-enclave code.
- □ V8. The memory hierarchy, specifically the row buffers are shared.

Summary of Attack vectors

- □ V1. Shared TLB entries under HT.
- □ V2. Selective TLB entries flushing without HT.
- □ V3. Referenced PTEs are cached as data.
- □ V4. Updates of accessed flags.
- □ V5. Updates of dirty flags.
- □ V6. Triggering page faults with P/X or reserved bits.
- □ V7. CPU caches are shared between the enclave and non-enclave code.
- □ V8. The memory hierarchy, specifically the row buffers are shared.

Summary of Attack vectors

- V1. Shared TLB entries under HT.
- V2. Selective TLB entries flushing without HT.
- V3. Referenced PTEs are cached as data.
- V4. Updates of accessed flags.
- V5. Updates of dirty flags.
- V6. Triggering page faults with P/X or reserved bits.
- V7. CPU caches are shared between the enclave and non-enclave code.
- V8. The memory hierarchy, specifically the row buffers are shared.

Summary of Attack vectors

- V1. Shared TLB entries under HT.
- V2. Selective TLB entries flushing without HT.
- V3. Referenced PTEs are cached as data.
- V4. Updates of accessed flags.
- V5. Updates of dirty flags.
- V6. Triggering page faults with P/X or reserved bits.
- V7. CPU caches are shared between the enclave and non-enclave code.
- V8. The memory hierarchy, specifically the row buffers are shared.

Summary of Attack vectors

- V1. Shared TLB entries under HT.
- V2. Selective TLB entries flushing without HT.
- V3. Referenced PTEs are cached as data.
- V4. Updates of accessed flags.
- V5. Updates of dirty flags.
- V6. Triggering page faults with P/X or reserved bits.
- V7. CPU caches are shared between the enclave and non-enclave code.
- V8. The memory hierarchy, specifically the row buffers are shared.

Summary of Attack vectors

- V1. Shared TLB entries under HT.
- V2. Selective TLB entries flushing without HT.
- V3. Referenced PTEs are cached as data.
- V4. Updates of accessed flags.
- V5. Updates of dirty flags.
- V6. Triggering page faults with P/X or reserved bits.
- V7. CPU caches are shared between the enclave and non-enclave code.
- V8. The memory hierarchy, specifically the row buffers are shared.

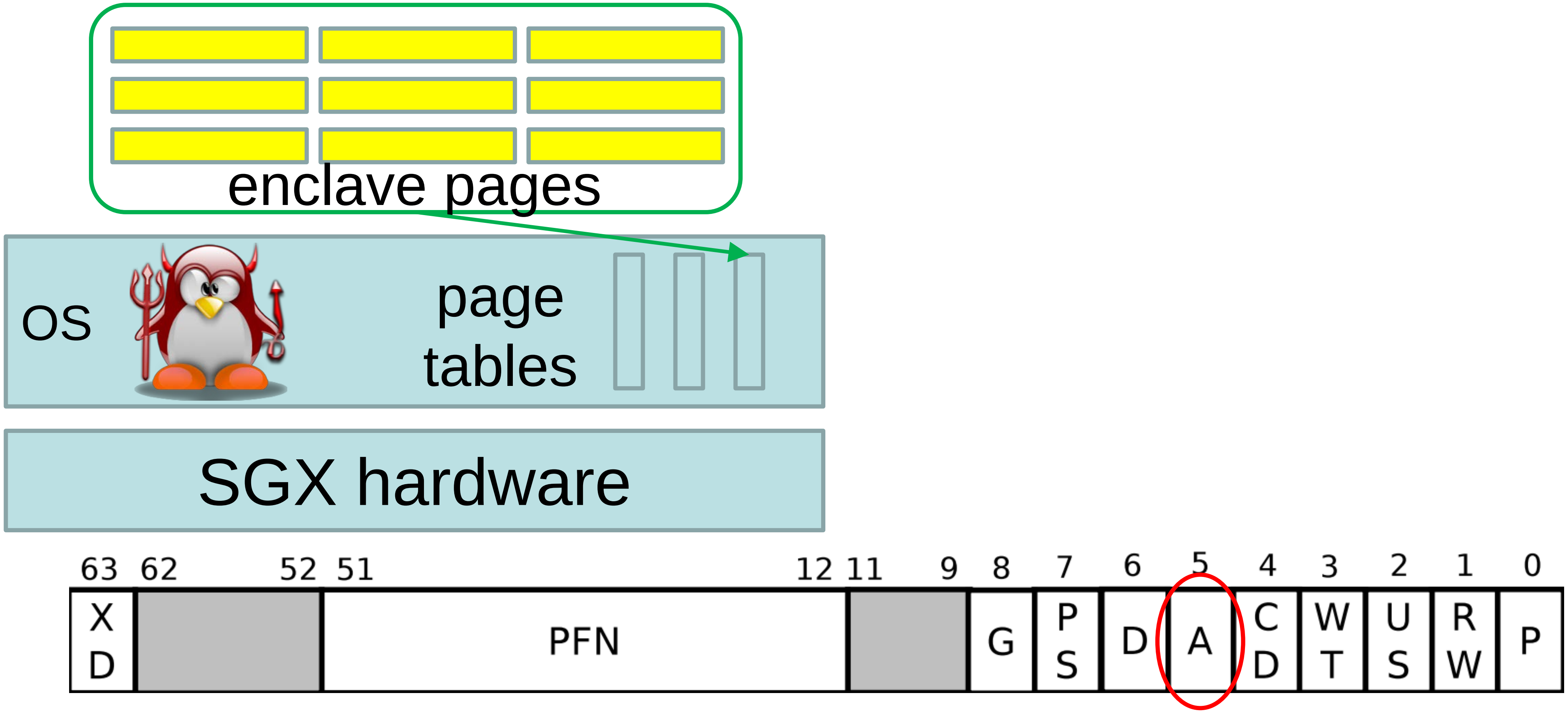
Can we make the attack stealthy by reducing AEXs induced by the attack?

2. Sneaky Page Monitoring Attacks (Vector 4)

- V1. Shared TLB entries under HT.
- V2. Selective TLB entries flushing without HT.
- V3. Referenced PTEs are cached as data.
- V4. Updates of accessed flags.
- V5. Updates of dirty flags.
- V6. Triggering page faults with P/X or reserved bits.
- V7. CPU caches are shared between the enclave and non-enclave code.
- V8. The memory hierarchy, specifically the row buffers are shared.

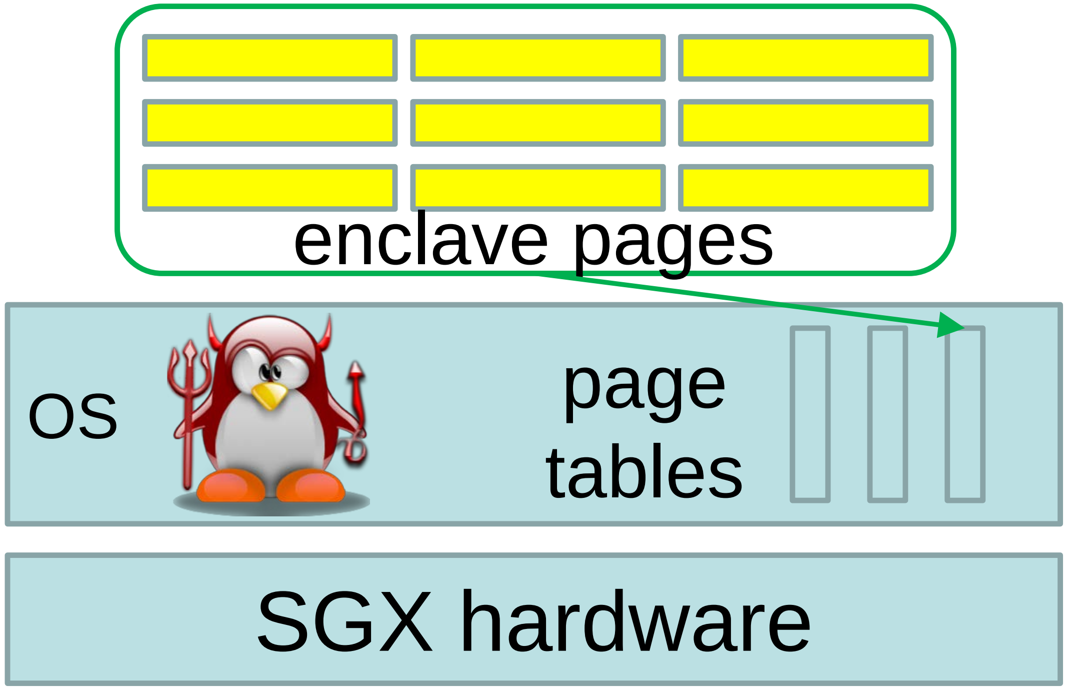
2. Sneaky Page Monitoring Attacks (Vector 4)

V4. Updates of accessed flags.



2. Sneaky Page Monitoring Attacks (Vector 4)

V4. Updates of accessed flags.

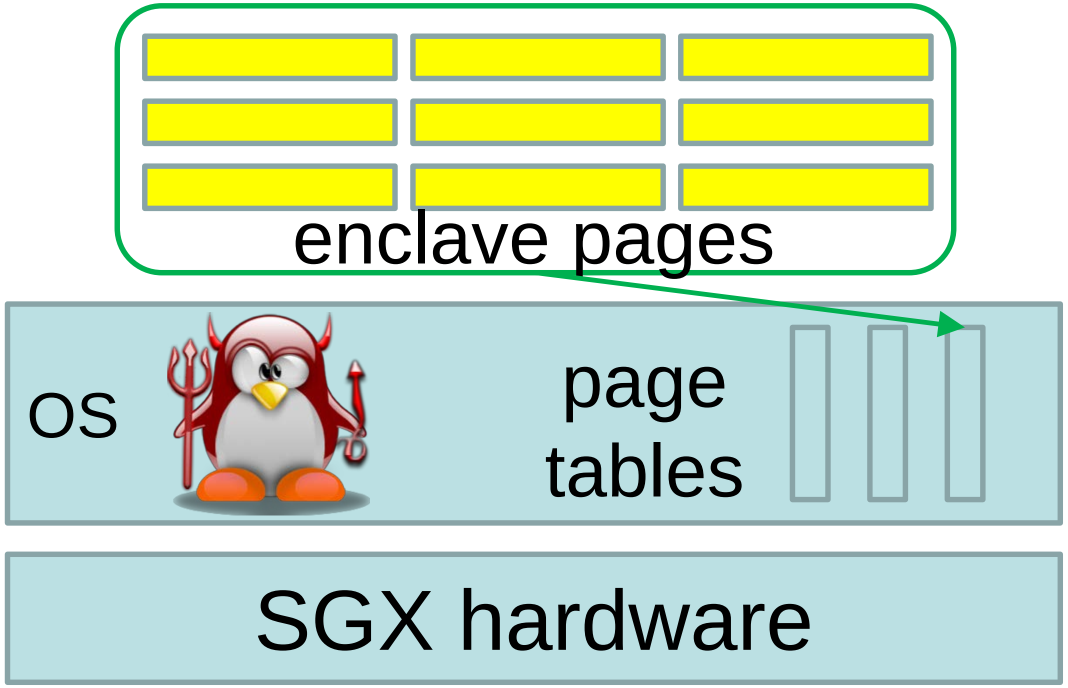


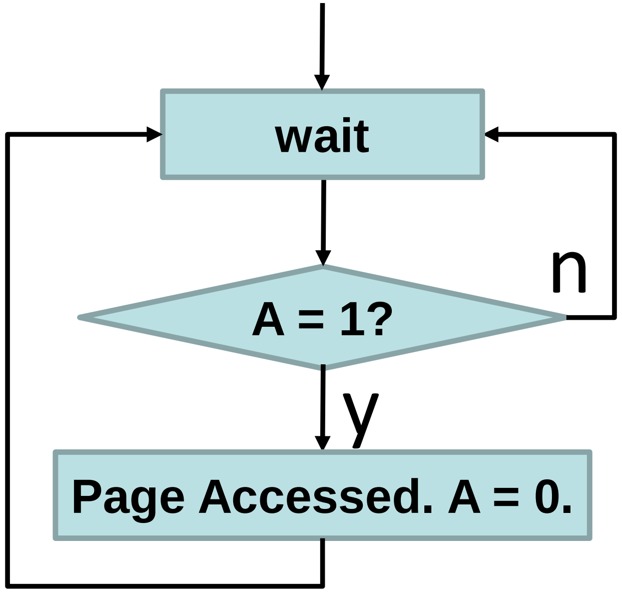
“Whenever the processor uses a paging-structure entry as part of linear-address translation, it sets the accessed flag in that entry (if it is not already set).”

63	62 52	51 12	11 9	8	7	6	5	4	3	2	1	0
XD		PFN		G	PS	D	A	CD	WT	US	RW	P

2. Sneaky Page Monitoring Attacks (Vector 4)

Basic accessed flags monitoring attack: B-SPM

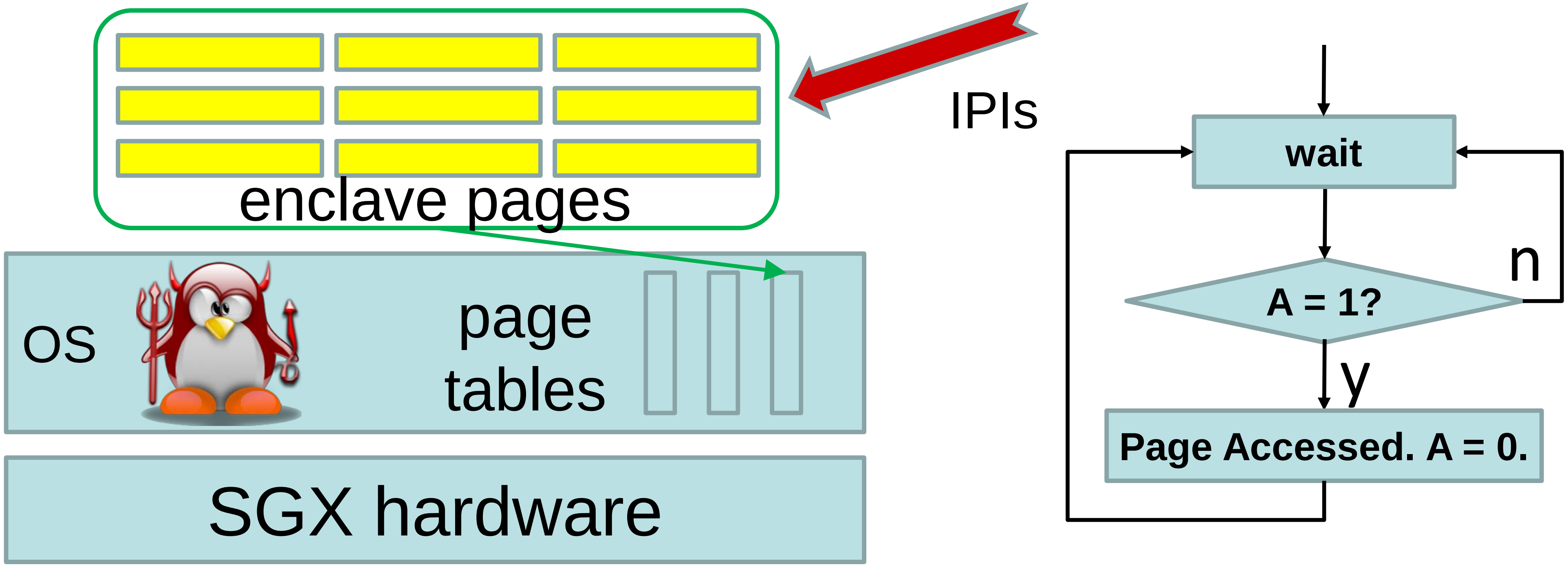




63	62 52	51 12	11 9	8	7	6	5	4	3	2	1	0
XD		PFN		G	PS	D	A	CD	WT	US	RW	P

2. Sneaky Page Monitoring Attacks (Vector 4)

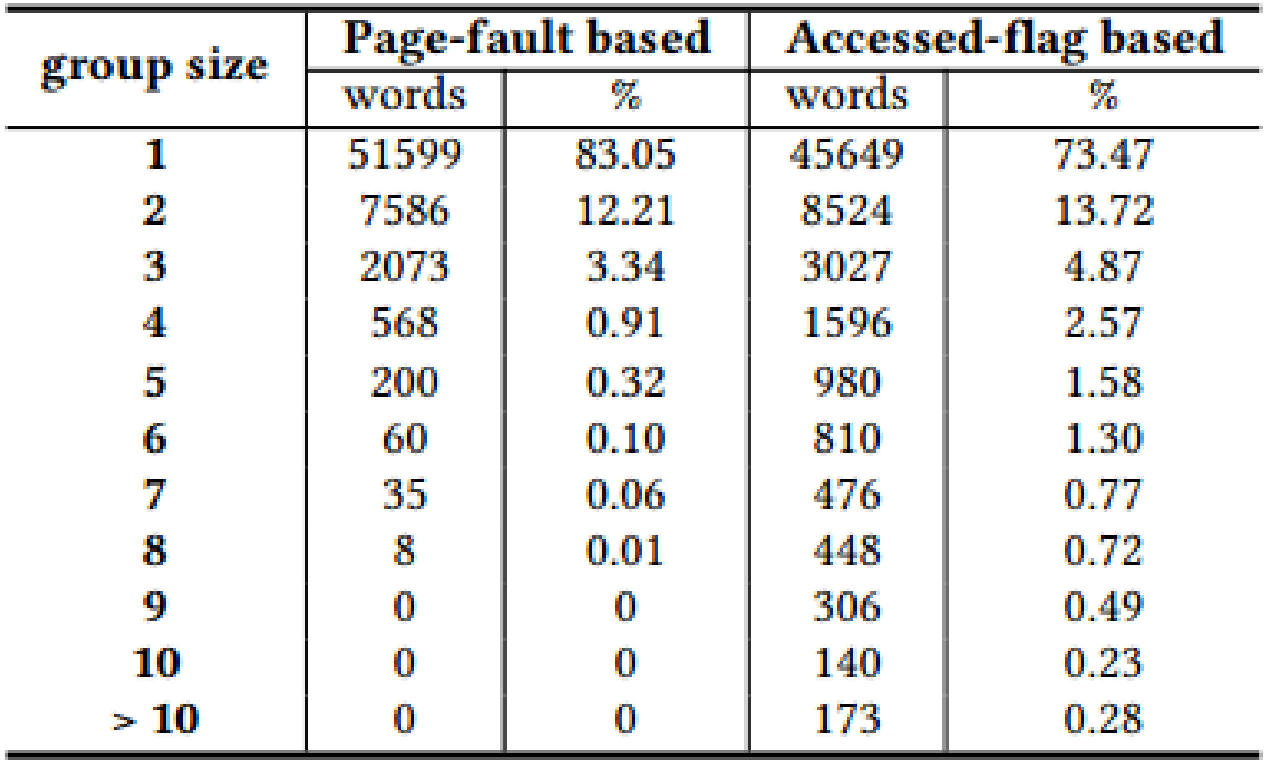
Basic accessed flags monitoring attack: B-SPM



63	62 … 52	51 … 12	11 … 9	8	7	6	5	4	3	2	1	0
XD		PFN		G	PS	D	A	CD	WT	US	RW	P

2. Sneaky Page Monitoring Attacks

Basic accessed flags monitoring attack: B-SPM



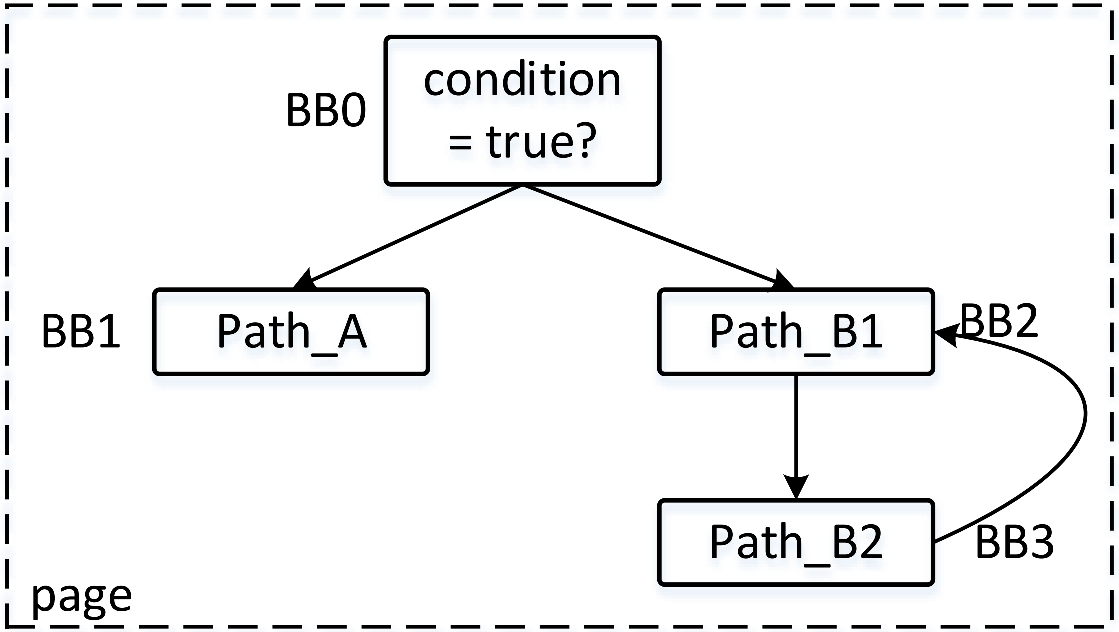
group size	Page-fault based		Accessed-flag based	
	words	%	words	%
1	51599	83.05	45649	73.47
2	7586	12.21	8524	13.72
3	2073	3.34	3027	4.87
4	568	0.91	1596	2.57
5	200	0.32	980	1.58
6	60	0.10	810	1.30
7	35	0.06	476	0.77
8	8	0.01	448	0.72
9	0	0	306	0.49
10	0	0	140	0.23
> 10	0	0	173	0.28

Evaluate on Hunspell.

Slowdown is brought down from 1214.9× for page fault attack to 5.1× for B-SPM attack.

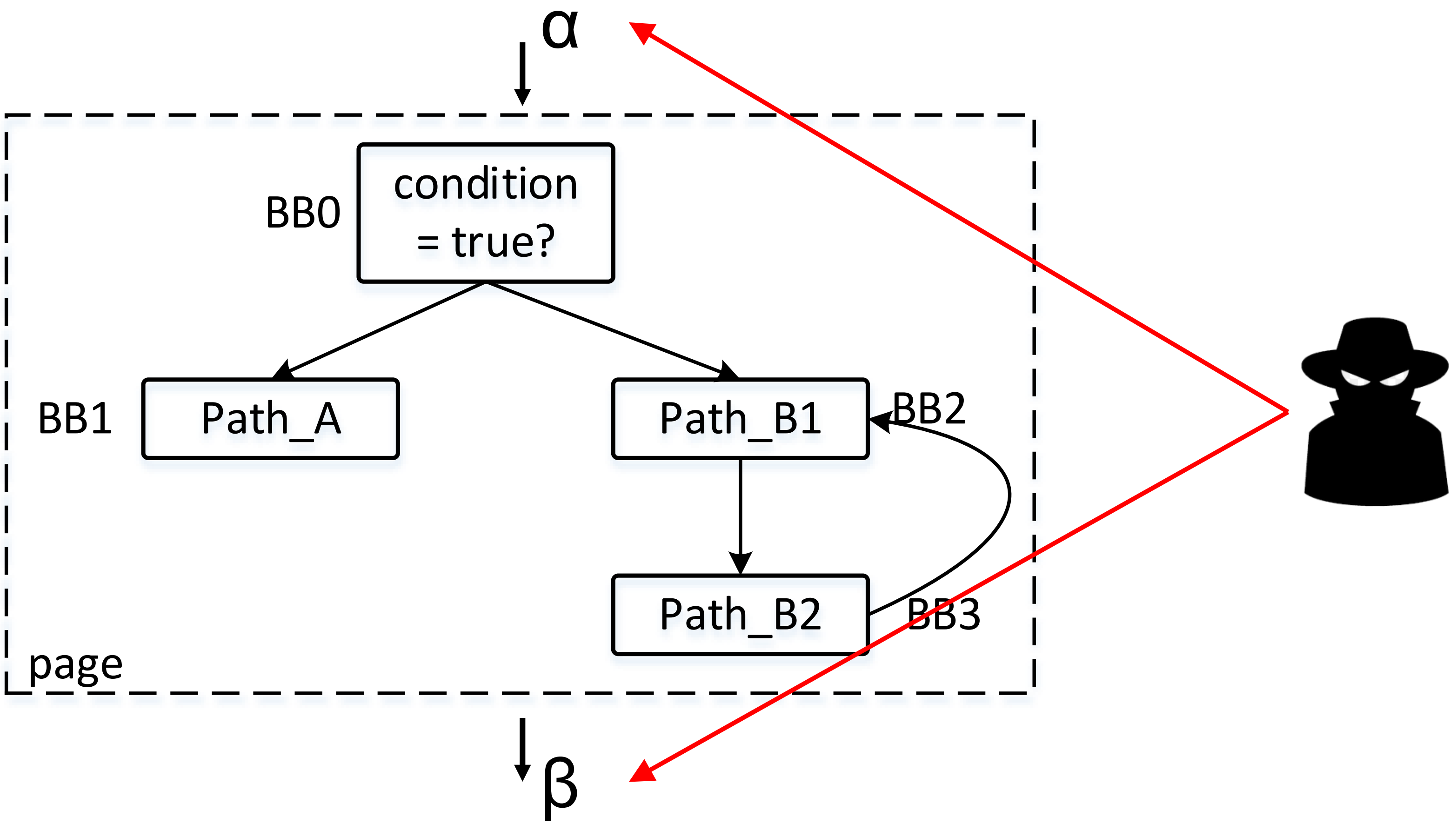
2. Sneaky Page Monitoring Attacks

What about if the pages that frequently accessed are to be observed?



2. Sneaky Page Monitoring Attacks

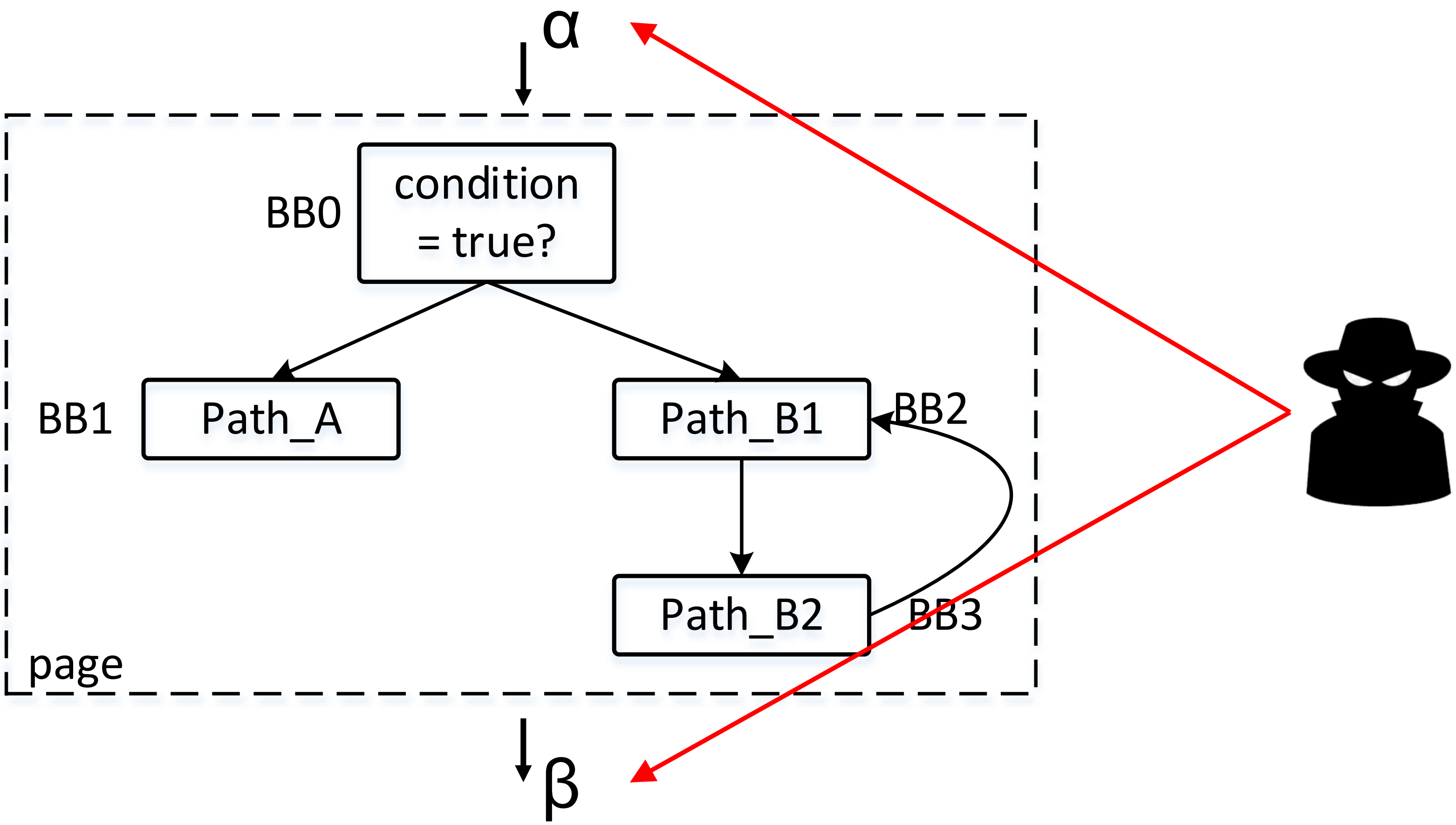
Timing enhancement: T-SPM



Measure the time between two monitored pages

2. Sneaky Page Monitoring Attacks

Timing enhancement: T-SPM



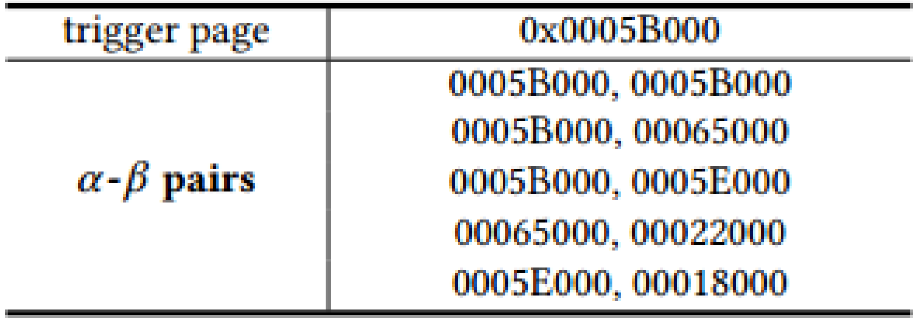
α_1	β_1	t_1
α_2	β_2	t_2
α_3	β_3	t_3
α_4	β_4	t_4
…	…	…

2. Sneaky Page Monitoring Attacks

Timing enhancement: T-SPM

Evaluate on FreeType.

Slowdown is brought down from 252× for page fault attack to 0.16× for T-SPM attack.



trigger page	0x0005B000
α-β **pairs**	0005B000, 0005B000
	0005B000, 00065000
	0005B000, 0005E000
	00065000, 00022000
	0005E000, 00018000

2. Sneaky Page Monitoring Attacks

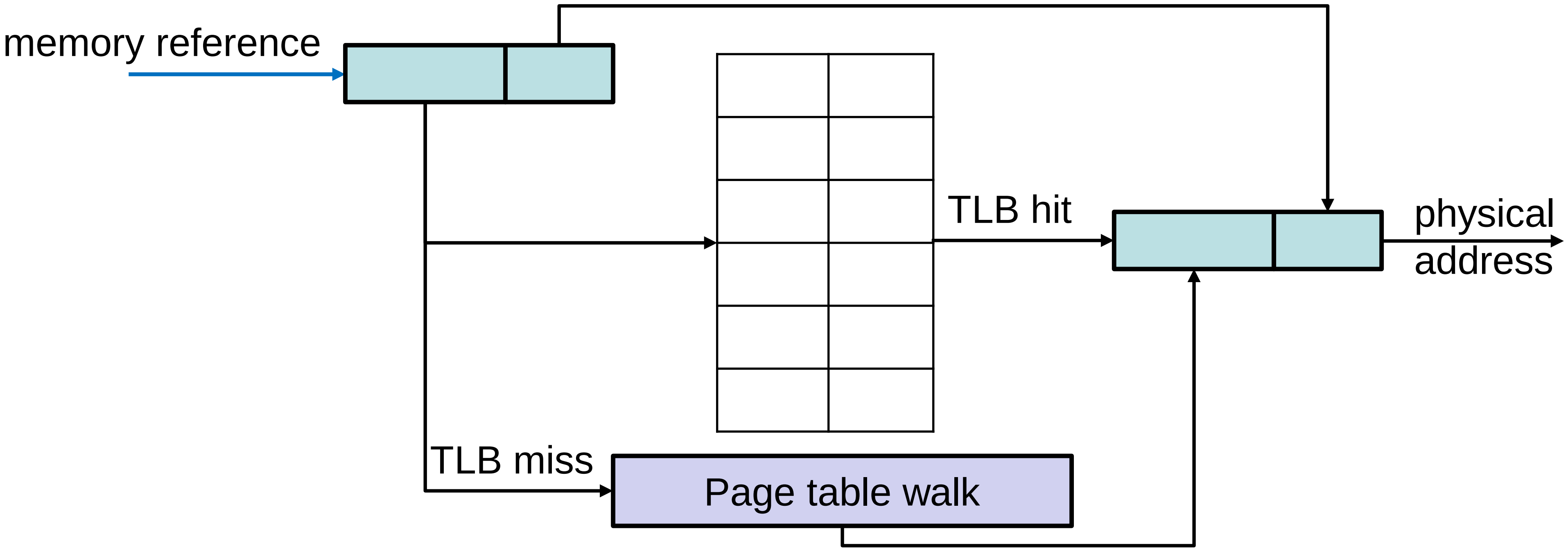
Can the side effect be further reduced?

2. Sneaky Page Monitoring Attacks

- V1. Shared TLB entries under HT.
- V2. Selective TLB entries flushing without HT.
- V3. Referenced PTEs are cached as data.
- V4/5. Updates of accessed/dirty flags.
- V6. Triggering page faults with P/X or reserved bits.
- V7. CPU caches are shared between the enclave and non-enclave code.
- V8. The memory hierarchy, specifically the row buffers are shared.

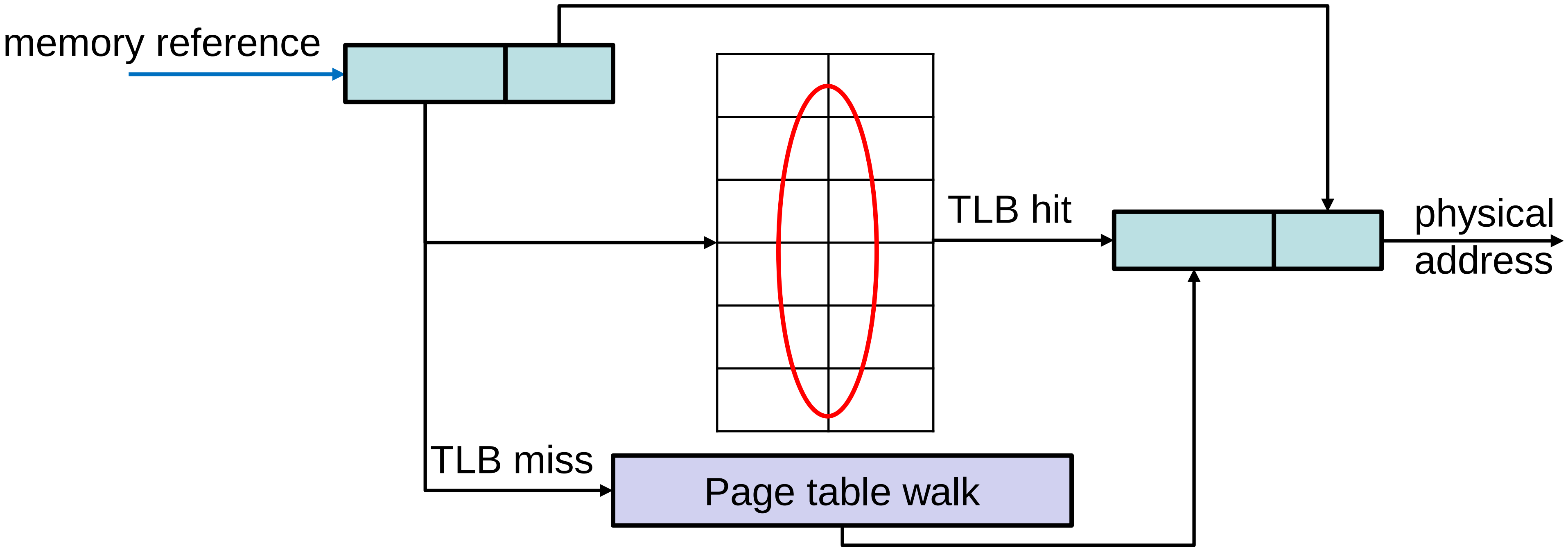
2. Sneaky Page Monitoring Attacks

TLB flushing with HT (Vector 1): HT-SPM



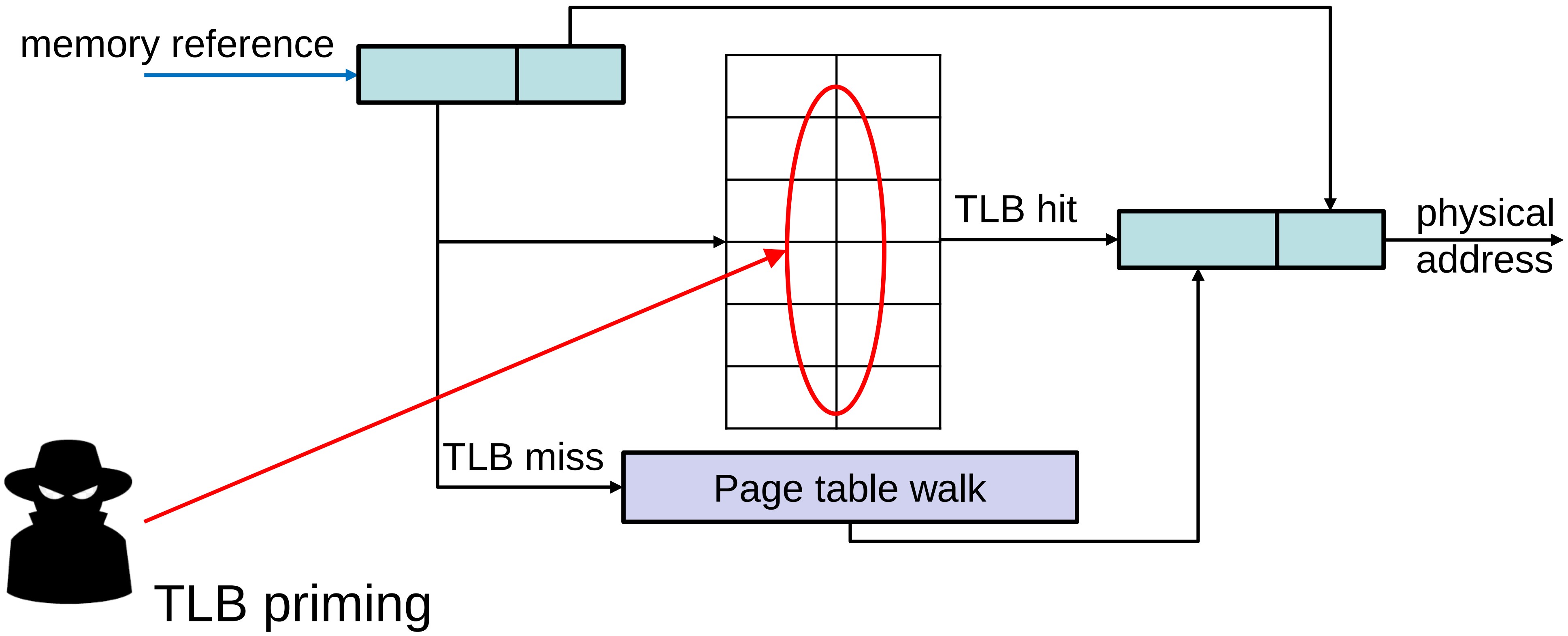
2. Sneaky Page Monitoring Attacks

TLB flushing with HT (Vector 1): HT-SPM



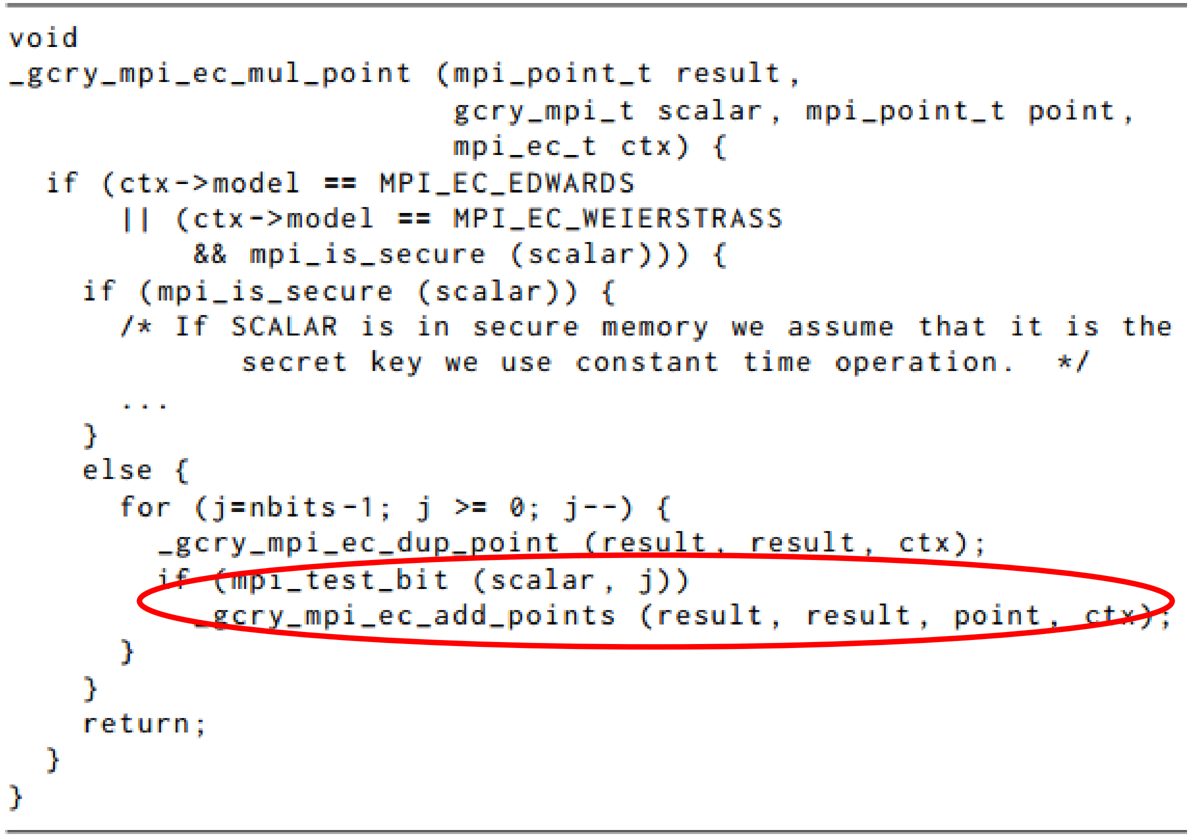
2. Sneaky Page Monitoring Attacks

TLB flushing with HT (Vector 1): HT-SPM



2. Sneaky Page Monitoring Attacks

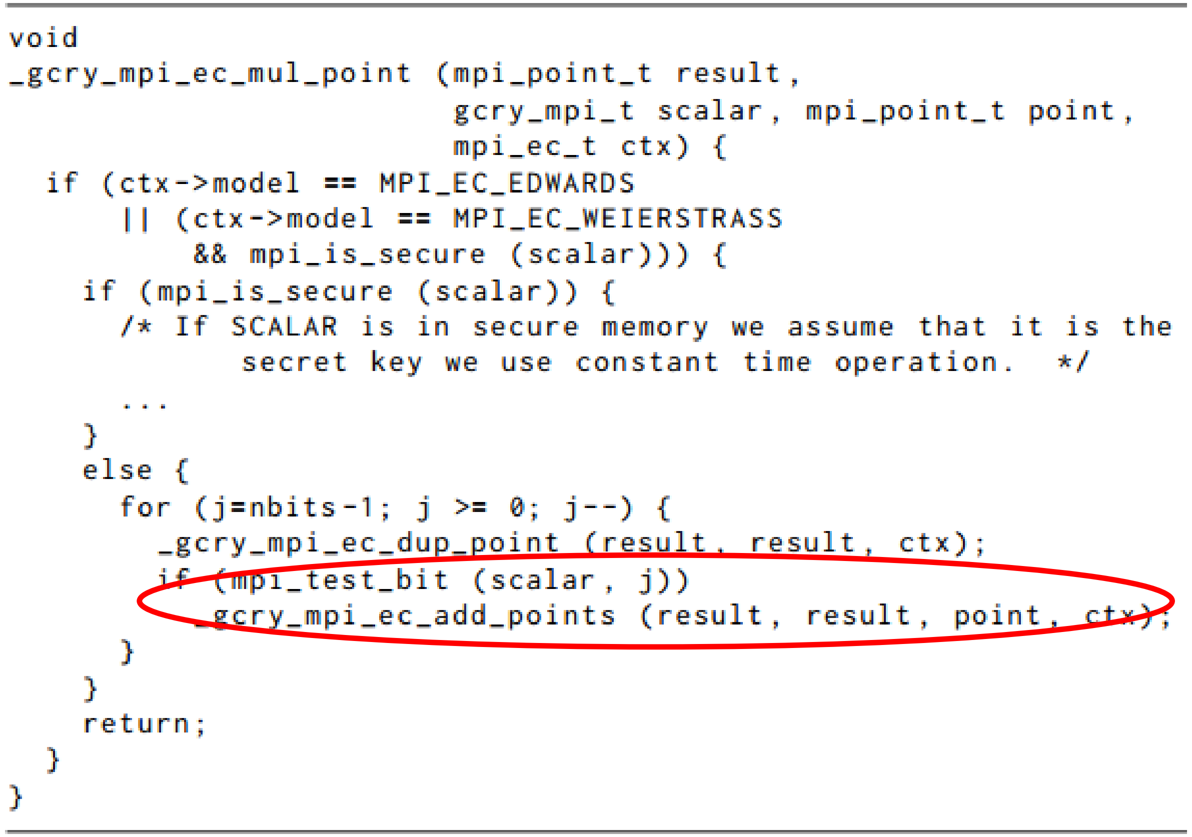
Evaluation on EdDSA of Libgcrypt v1.7.6



```
void
_gcry_mpi_ec_mul_point (mpi_point_t result,
                        gcry_mpi_t scalar, mpi_point_t point,
                        mpi_ec_t ctx) {
  if (ctx->model == MPI_EC_EDWARDS
      || (ctx->model == MPI_EC_WEIERSTRASS
          && mpi_is_secure (scalar))) {
    if (mpi_is_secure (scalar)) {
      /* If SCALAR is in secure memory we assume that it is the
           secret key we use constant time operation.  */
      ...
    }
    else {
      for (j=nbits-1; j >= 0; j--) {
        _gcry_mpi_ec_dup_point (result, result, ctx);
        if (mpi_test_bit (scalar, j))
          _gcry_mpi_ec_add_points (result, result, point, ctx);
      }
    }
    return;
  }
}
```

2. Sneaky Page Monitoring Attacks

Evaluation on EdDSA of Libgcrypt v1.7.6



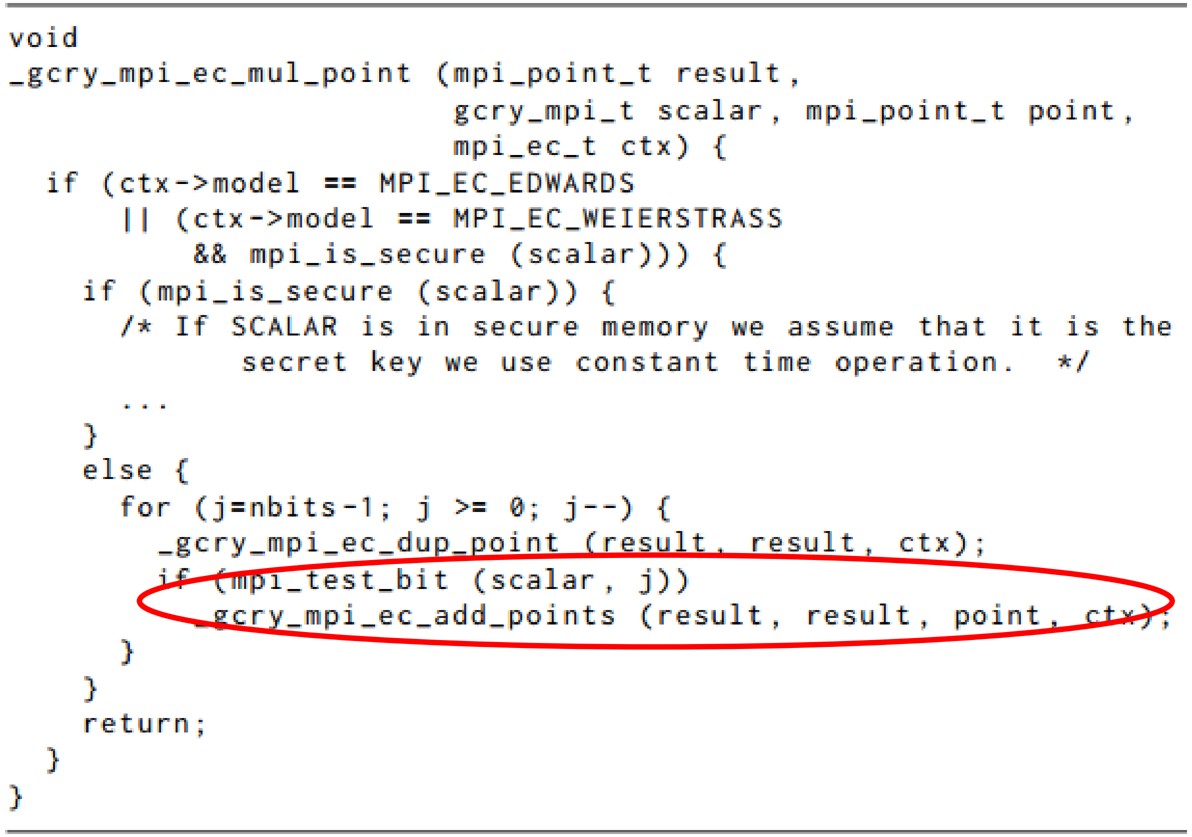
```
void
_gcry_mpi_ec_mul_point (mpi_point_t result,
            gcry_mpi_t scalar, mpi_point_t point,
            mpi_ec_t ctx) {
 if (ctx->model == MPI_EC_EDWARDS
   || (ctx->model == MPI_EC_WEIERSTRASS
     && mpi_is_secure (scalar))) {
  if (mpi_is_secure (scalar)) {
   /* If SCALAR is in secure memory we assume that it is the
      secret key we use constant time operation.  */
   ...
  }
  else {
   for (j=nbits-1; j >= 0; j--) {
    _gcry_mpi_ec_dup_point (result, result, ctx);
    if (mpi_test_bit (scalar, j))
     _gcry_mpi_ec_add_points (result, result, point, ctx);
   }
  }
  return;
 }
}
```

Attacks	Number of AEXs
Page fault attack	71,000
B-SPM attack	33,000
T-SPM attack	1,300

* HT-SPM is designed to reduce AEXs for data pages, and is not presented in the comparison.

2. Sneaky Page Monitoring Attacks

Evaluation on EdDSA of Libgcrypt v1.7.6



```
void
_gcry_mpi_ec_mul_point (mpi_point_t result,
                        gcry_mpi_t scalar, mpi_point_t point,
                        mpi_ec_t ctx) {
  if (ctx->model == MPI_EC_EDWARDS
      || (ctx->model == MPI_EC_WEIERSTRASS
          && mpi_is_secure (scalar))) {
    if (mpi_is_secure (scalar)) {
      /* If SCALAR is in secure memory we assume that it is the
             secret key we use constant time operation.  */
      ...
    }
    else {
      for (j=nbits-1; j >= 0; j--) {
        _gcry_mpi_ec_dup_point (result, result, ctx);
        if (mpi_test_bit (scalar, j))
          _gcry_mpi_ec_add_points (result, result, point, ctx);
      }
    }
    return;
  }
}
```

Attacks	Number of AEXs
Page fault attack	71,000
B-SPM attack	33,000
T-SPM attack	1,300

* HT-SPM is designed to reduce AEXs for data pages, and is not presented in the comparison.

3. Achieving fine-grained spatial granularity

- Cache-based attack
  - Prime+Probe: 16 KB, if 2048 cache set, 128 MB EPC
  - ~~Flush+Reload: 64 B~~

3. Achieving fine-grained spatial granularity

- Cache-based attack
  - Prime+Probe: 16 KB, if 2048 cache set, 128 MB EPC
  - ~~Flush+Reload: 64 B~~
- DRAMA attack
  - The program needs to have a large memory footprint, otherwise the memory reference will mostly hit the cache.

3. Achieving fine-grained spatial granularity

- Cache-based attack
  - Prime+Probe: 16 KB, if 2048 cache set, 128 MB EPC
  - ~~Flush+Reload: 64 B~~
- DRAMA attack
  - The program needs to have a large memory footprint, otherwise the memory reference will mostly hit the cache.

Cache-DRAM attack: finer-grained attack with less noise.

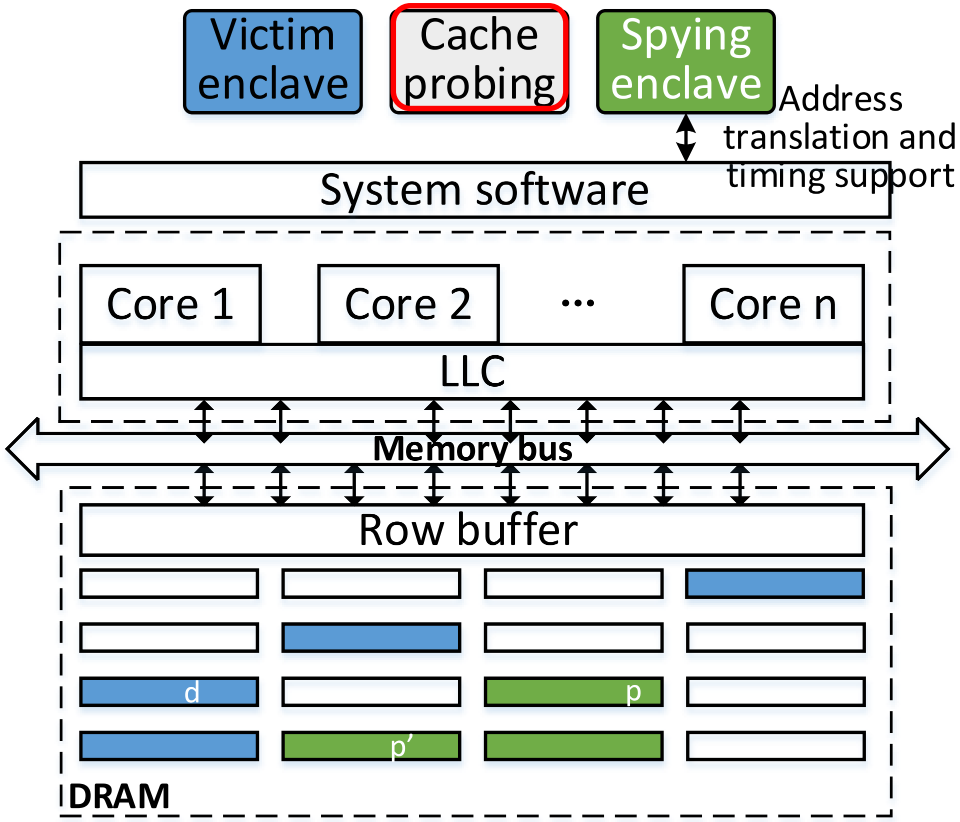
3. Achieving fine-grained spatial granularity

Cache-DRAM attack

64 B granularity

- ❑ DRAM rows are only shared among enclaves.
- ❑ No high resolution timer inside the enclaves.

cache priming



3. Achieving fine-grained spatial granularity

Cache-DRAM attack

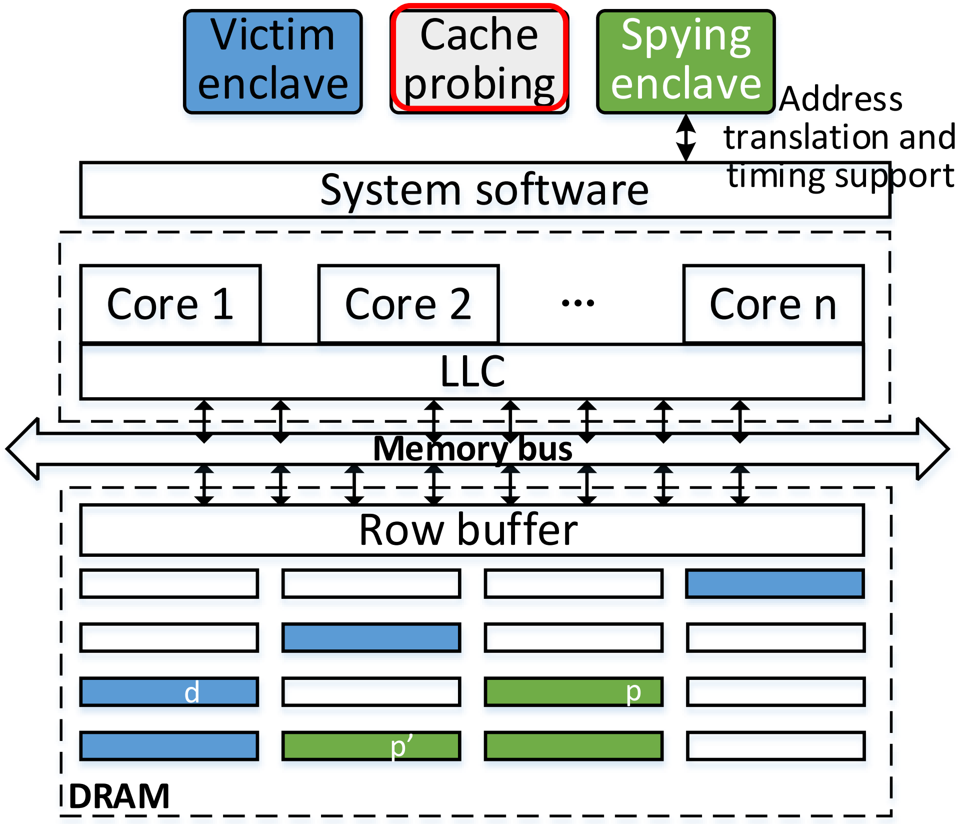
64 B granularity

- ❑ DRAM rows are only shared among enclaves.
- ❑ No high resolution timer inside the enclaves.

Evaluation on a conditional branch in Gap 4.8.6.

14.6% detection, <1% false detection.

cache priming



Summary of Attack Vectors

Vectors	**Spatial granularity**	**AEX**	**Slow-down**
* i/dCache PRIME+PROBE	2 MB	High	High
* L2 Cache PRIME+PROBE	128 KB	High	High
L3 Cache PRIME+PROBE	16 KB	None	Modest
Page fault attack	4 KB	High	High
B/T-SPM	4 KB	Modest	Modest
HT-SPM	4 KB	None	Modest
Cross-enclave DRAMA	1 KB	None	High
Cache-DRAM	64 B	None	Minimal

* Do not consider attacks under HT. Otherwise the AEX and slow-down will be low.

Summary of Attack Vectors

Vectors	**Spatial granularity**	**AEX**	**Slow-down**
* i/dCache PRIME+PROBE	2 MB	High	High
* L2 Cache PRIME+PROBE	128 KB	High	High
L3 Cache PRIME+PROBE	16 KB	None	Modest
Page fault attack	4 KB	High	High
B/T-SPM	4 KB	Modest	Modest
HT-SPM	4 KB	None	Modest
Cross-enclave DRAMA	1 KB	None	High
Cache-DRAM	64 B	None	Minimal

* Do not consider attacks under HT. Otherwise the AEX and slow-down will be low.

Summary of Attack Vectors

Vectors	**Spatial granularity**	**AEX**	**Slow-down**
* i/dCache PRIME+PROBE	2 MB	High	High
* L2 Cache PRIME+PROBE	128 KB	High	High
L3 Cache PRIME+PROBE	16 KB	None	Modest
Page fault attack	4 KB	High	High
B/T-SPM	4 KB	Modest	Modest
HT-SPM	4 KB	None	Modest
Cross-enclave DRAMA	1 KB	None	High
Cache-DRAM	64 B	None	Minimal

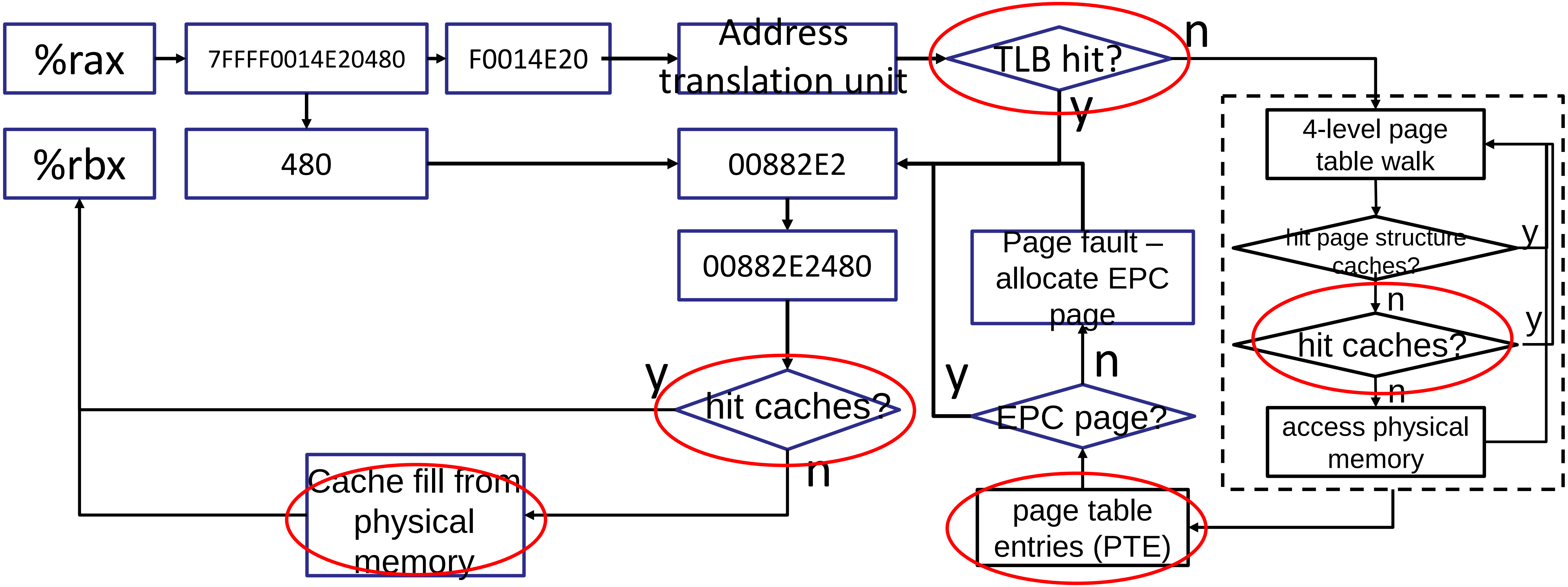
* Do not consider attacks under HT. Otherwise the AEX and slow-down will be low.

Conclusions

- We identified 8 attack vectors in SGX memory management.

Looking again at the attack surfaces

mov (%rax), %rbx



Conclusions

- We identified 8 attack vectors in SGX memory management.
  - There can be more.

Conclusions

- We identified 8 attack vectors in SGX memory management.
  - There can be more.
- New attacks that induce few AEXs, that bypass existing defenses

Conclusions

- We identified 8 attack vectors in SGX memory management.
  - There can be more.
- New attacks that induce few AEXs, that bypass existing defenses
  - Interrupts are not necessary to attack the enclave.

Conclusions

- We identified 8 attack vectors in SGX memory management.
  - There can be more.
- New attacks that induce few AEXs, that bypass existing defenses
  - Interrupts are not necessary to attack the enclave.
- Attacks can achieve finer-grained spatial granularity.

Conclusions

- We identified 8 attack vectors in SGX memory management.
  - There can be more.
- New attacks that induce few AEXs, that bypass existing defenses
  - Interrupts are not necessary to attack the enclave.
- Attacks can achieve finer-grained spatial granularity.
- Attack vectors can be combined to be more effective
  - TLB flushing + SPM, Cache + DRAM, Page monitoring + timing
  - Others?

Conclusions

- We identified 8 attack vectors in SGX memory management.
  - There can be more.
- New attacks that induce few AEXs, that bypass existing defenses
  - Interrupts are not necessary to attack the enclave.
- Attacks can achieve finer-grained spatial granularity.
- Attack vectors can be combined to be more effective
  - TLB flushing + SPM, Cache + DRAM, Page monitoring + timing
  - Others?
- Defenses?

Thanks! Any questions?

ww31@indiana.edu

Backup Slides

Characterizing memory vectors

Spatial granularity

The smallest unit of information directly observable to the adversary.

Temporal observability

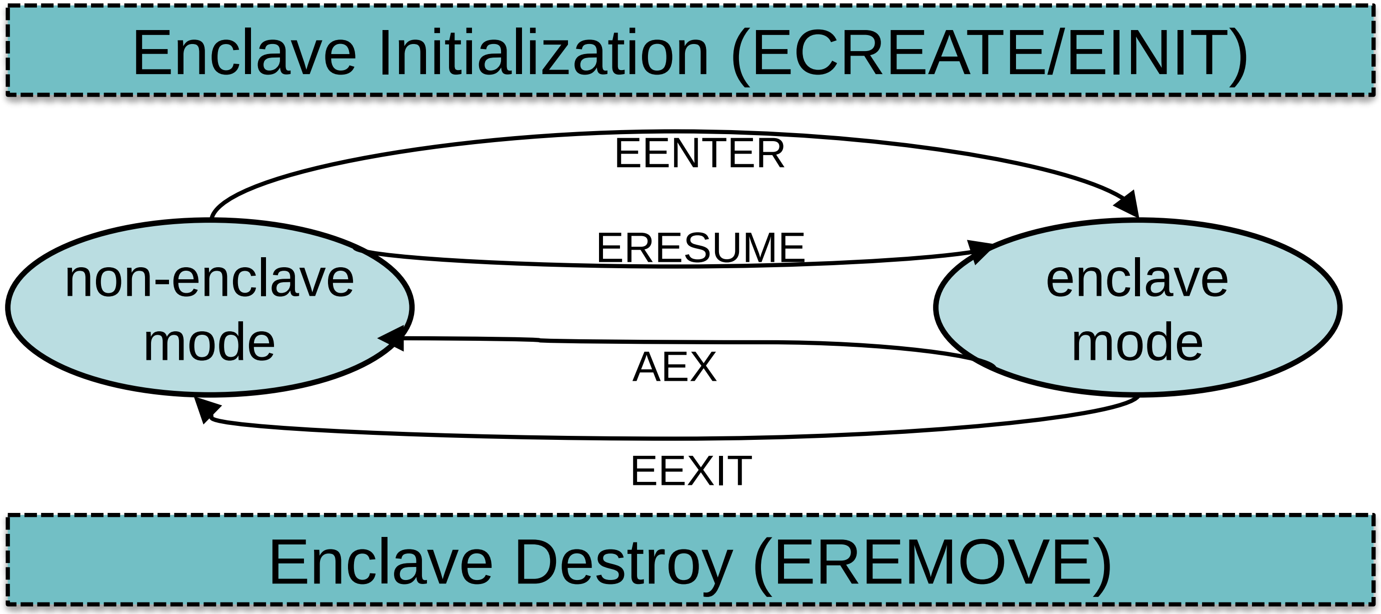
The ability for the adversary to measure the timing signals generated during the execution of the target program.

Side effects

Observable anomalies caused by an attack, which could be employed to detect the attack, such as AEX.

Intel Software Guard Extensions

- Life cycle of an enclave thread


Enclave Initialization (ECREATE/EINIT)
EENTER
ERESUME
non-enclave mode
enclave mode
AEX
EEXIT
Enclave Destroy (EREMOVE)

Related work on Security’17

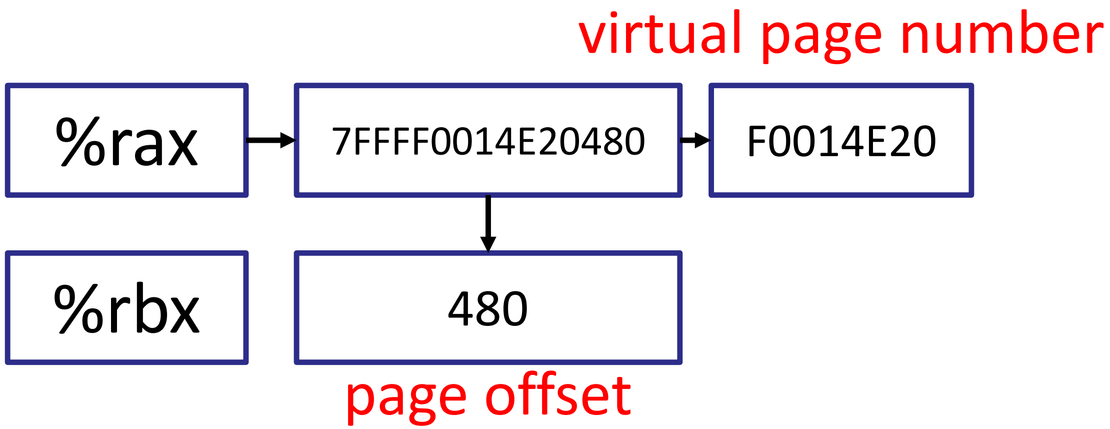
- Vector 3, 4

1. Understanding Attack Surfaces

```
mov (%rax), %rbx
```

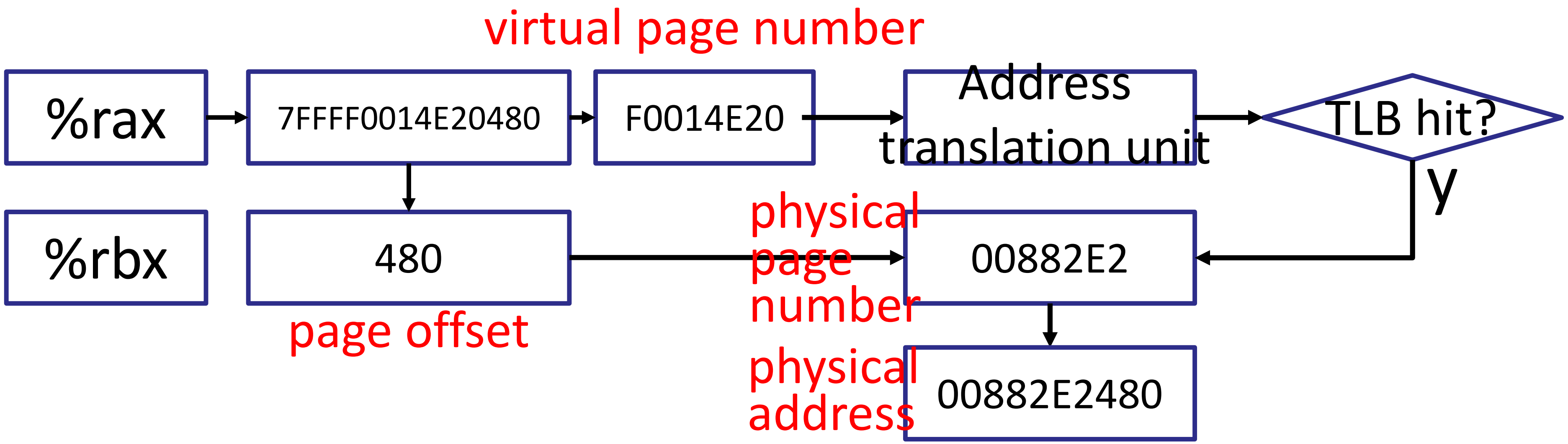
1. Understanding Attack Surfaces

```
mov (%rax), %rbx
```



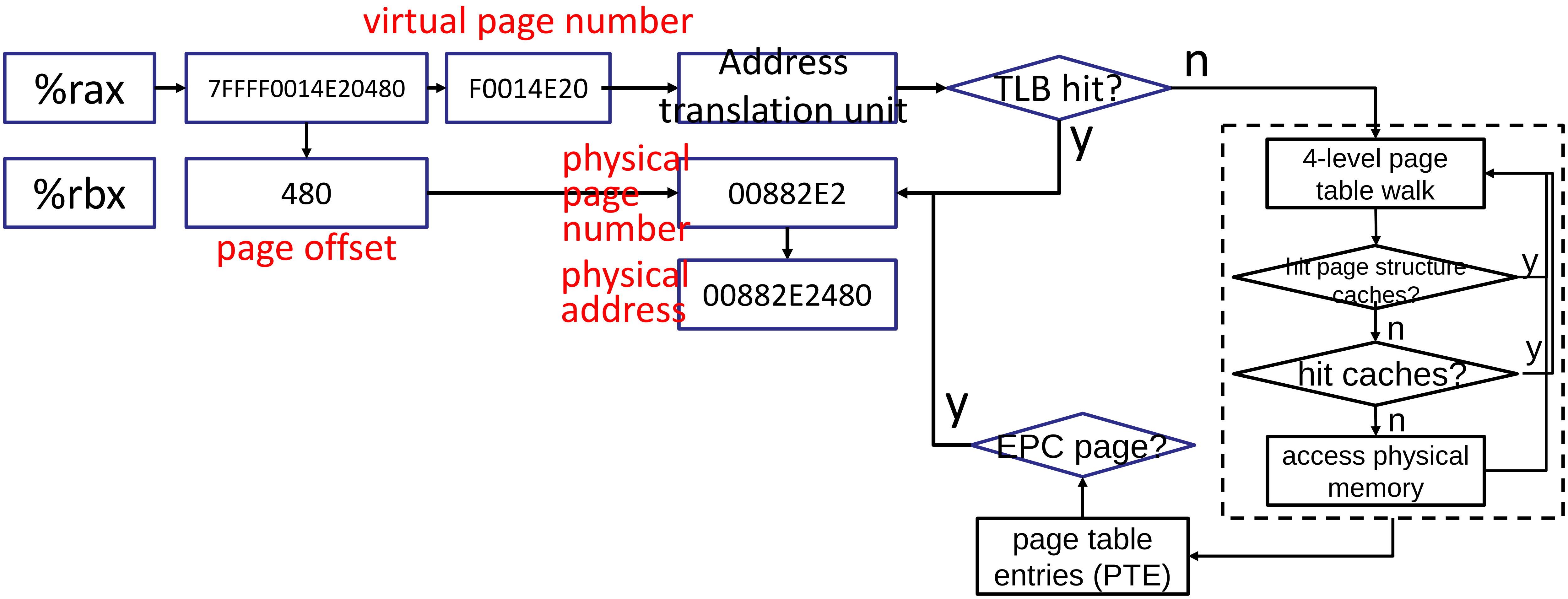
1. Understanding Attack Surfaces

mov (%rax), %rbx



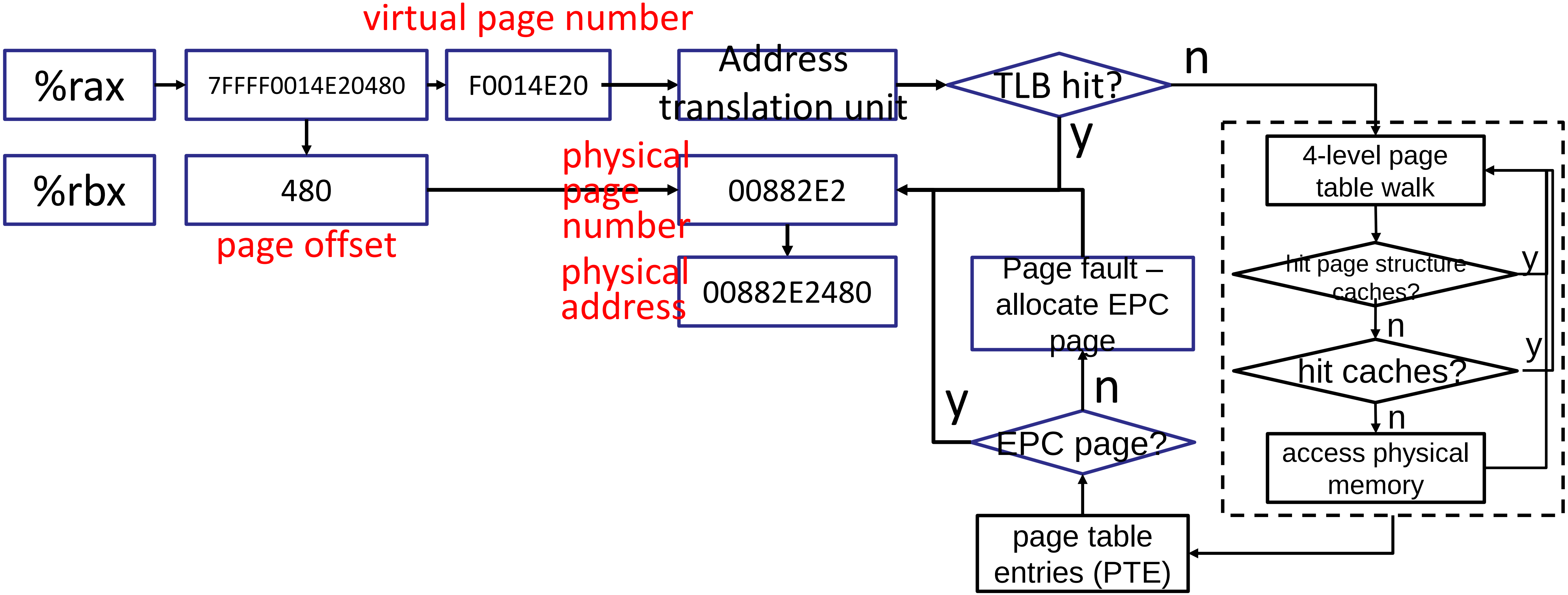
1. Understanding Attack Surfaces

mov (%rax), %rbx



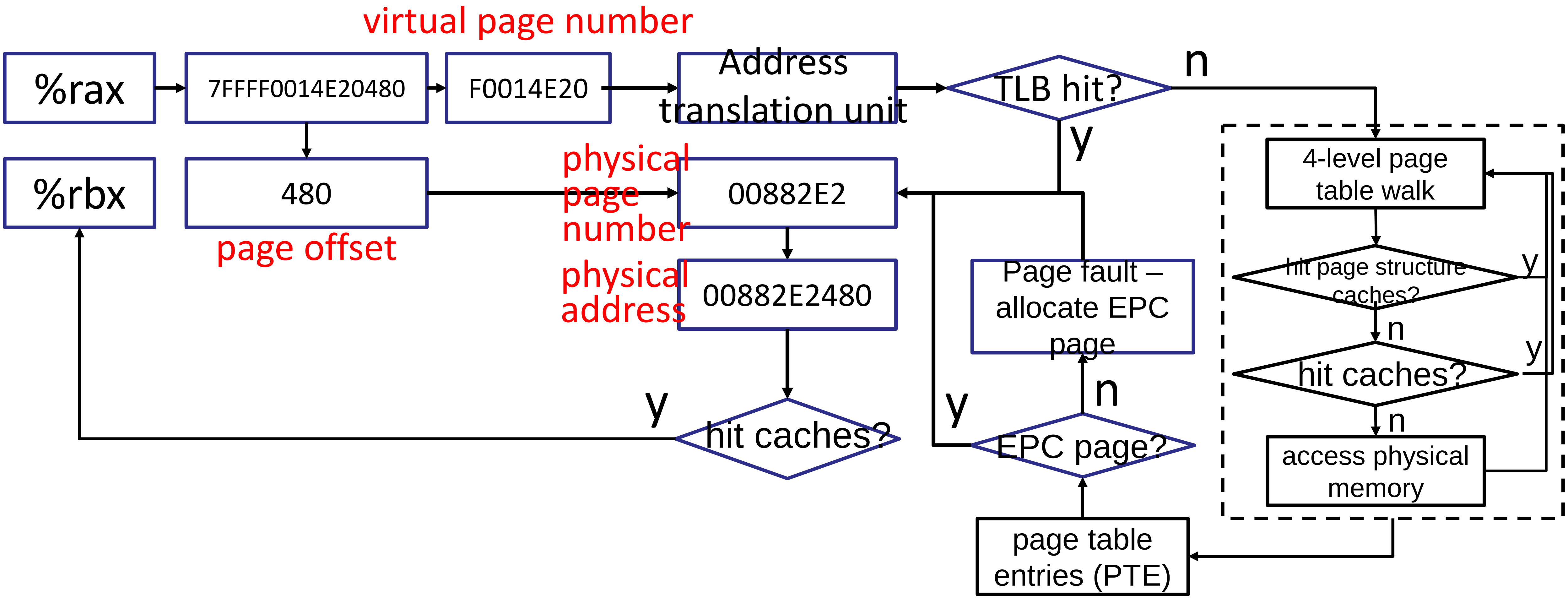
1. Understanding Attack Surfaces

mov (%rax), %rbx



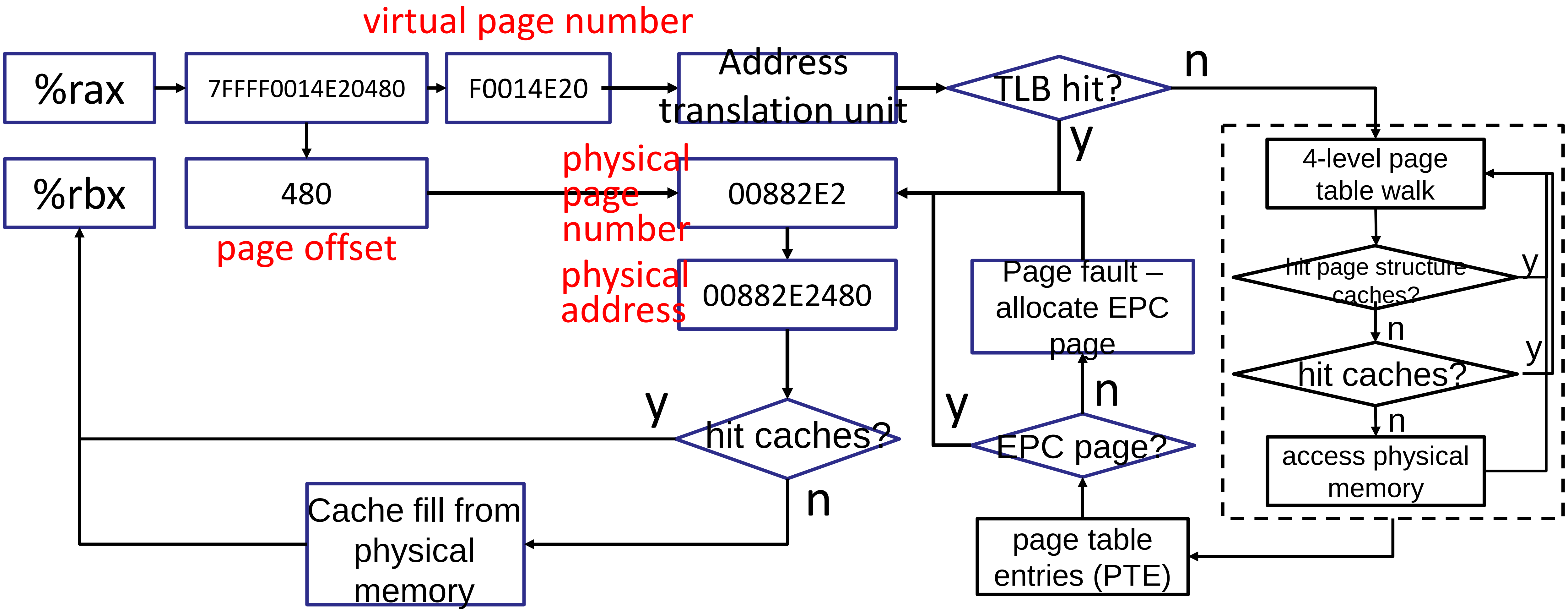
1. Understanding Attack Surfaces

```
mov (%rax), %rbx
```



1. Understanding Attack Surfaces

mov (%rax), %rbx


virtual page number
%rax
7FFFF0014E20480
F0014E20
Address translation unit
TLB hit?
n
y
%rbx
480
page offset
physical page number
00882E2
physical address
00882E2480
4-level page table walk
hit page structure caches?
y
n
hit caches?
y
n
access physical memory
Page fault – allocate EPC page
n
EPC page?
y
page table entries (PTE)
hit caches?
y
n
Cache fill from physical memory