Insights

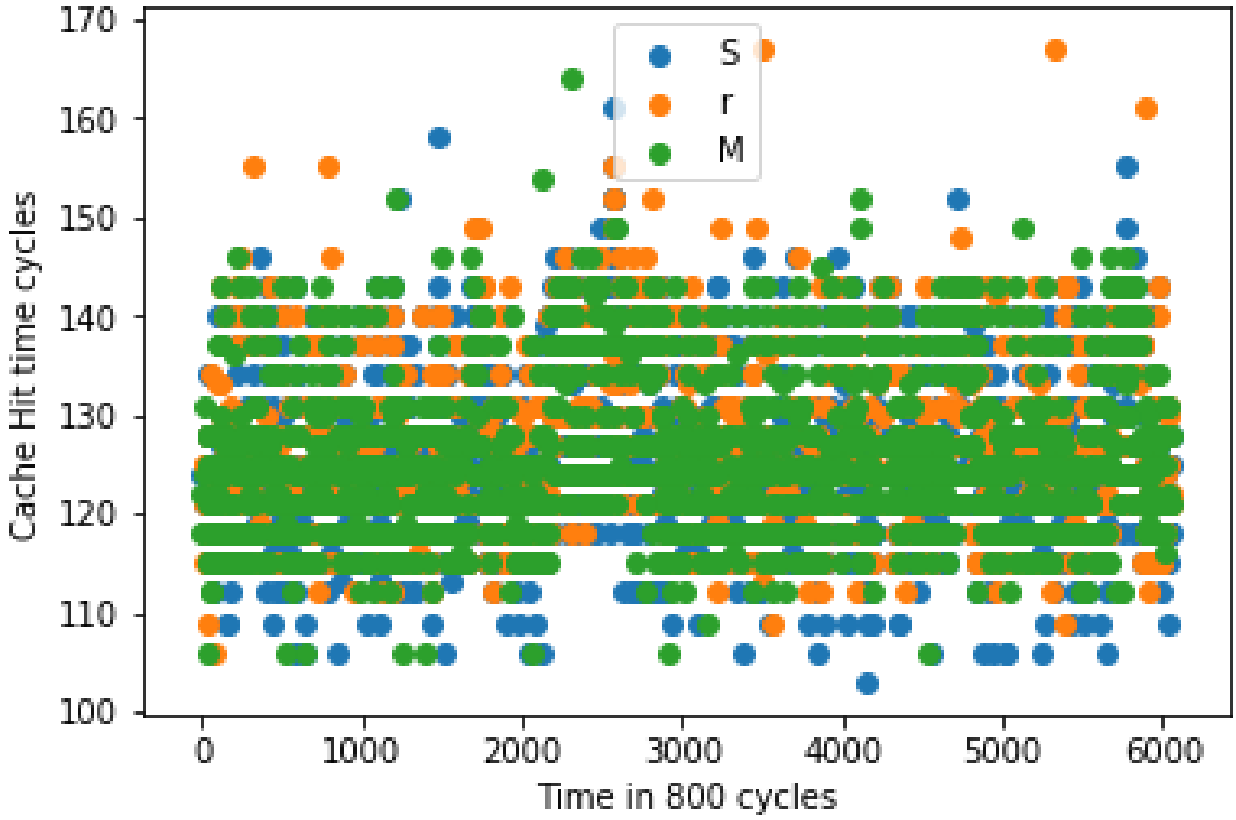
SYSTEM CONFIGURATION:

LEVEL1_ICACHE_SIZE	32768
LEVEL1_ICACHE_ASSOC	8
LEVEL1_ICACHE_LINESIZE	64
LEVEL1_DCACHE_SIZE	32768
LEVEL1_DCACHE_ASSOC	8
LEVEL1_DCACHE_LINESIZE	64
LEVEL2_CACHE_SIZE	262144
LEVEL2_CACHE_ASSOC	8
LEVEL2_CACHE_LINESIZE	64
LEVEL3_CACHE_SIZE	3145728
LEVEL3_CACHE_ASSOC	12
LEVEL3_CACHE_LINESIZE	64
LEVEL4_CACHE_SIZE	0
LEVEL4_CACHE_ASSOC	0
LEVEL4_CACHE_LINESIZE	0

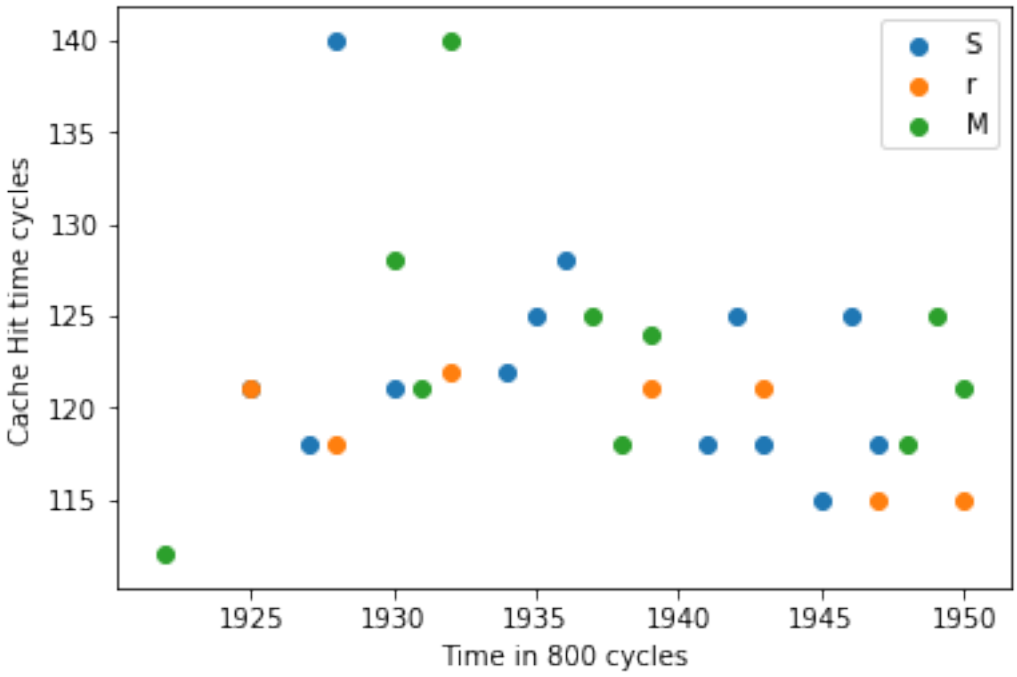
Flush and Reload

- We implemented flush and reload on our machine (HP AC045TU , Intel i5, 5th Gen) with above cache structure
- we attacked on GnuPG 1.4.13 by finding the addresses of Square , Reduce and Multiplication functions

Plot :



SnapShot of 24000 cycles:



Insights :

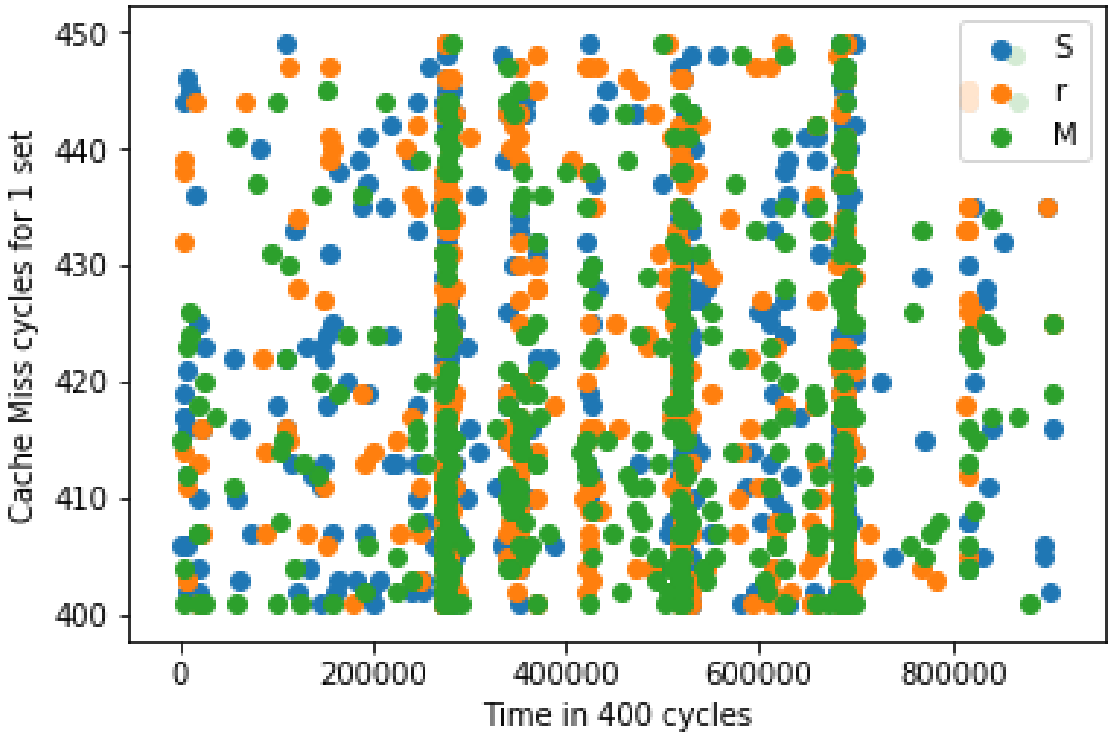
- We found below pattern while decrypting a file using GnuPG in our system by placing the printf satement in the square,reduce,and reduce functions.
  0 – SSSSr
  1 – SSSSrMMMMr
- We also got similar pattern for while attacking the GnuPG, but sometimes we got less ‘S’s and less ‘M’s(Eg: SSr , Sr ,MMr, Mr) becauseo of **overlapping** of the victim and attacker
- We lost some data because of the **concurrent execution** of victim and programmer which resulted overlapping of victim intervals and attacker intervals.
- We are getting different number of bits in different trials for the same interval because of following reasons.
  Overlapping intervals : both victim and attacker are trying to access the address at same time.
  Scheduling: As we know that different processes will be scheduled in same core, because of this victim and attacker are not scheduled at same time even though we are running them parallely
  When we run attacker programm with **high priority** we were able extract more bits.

Accuray :

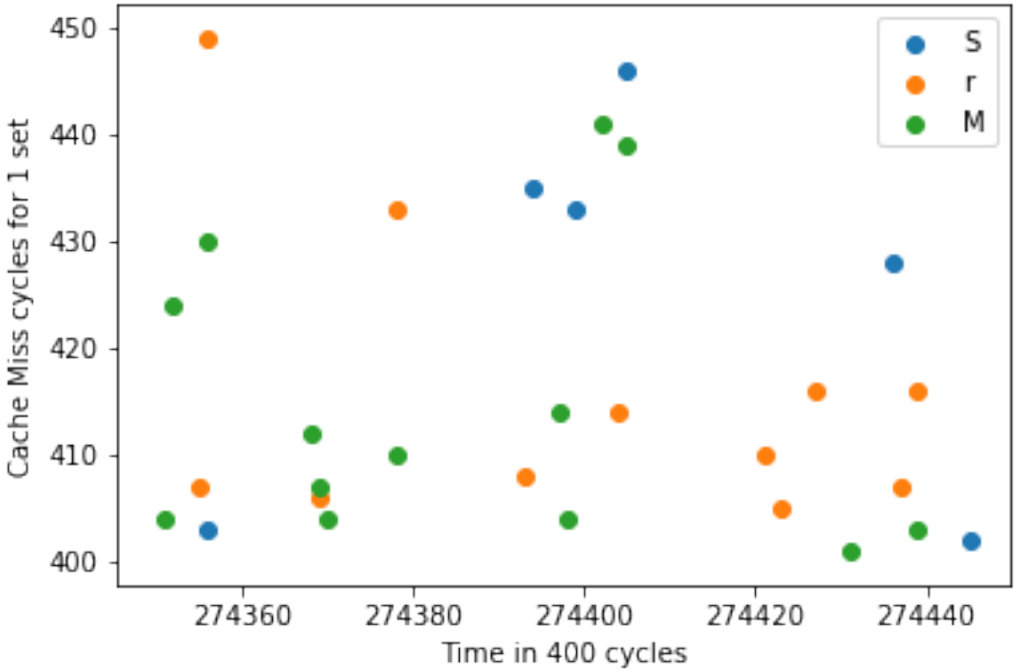
- We got 1009 bits at a particular instance , whose edit distance 1039. It is about 49 % accuracy.

Prime and Probe on L1 Instruction Cache:-

Plot:-



Snap click :-



Insights :

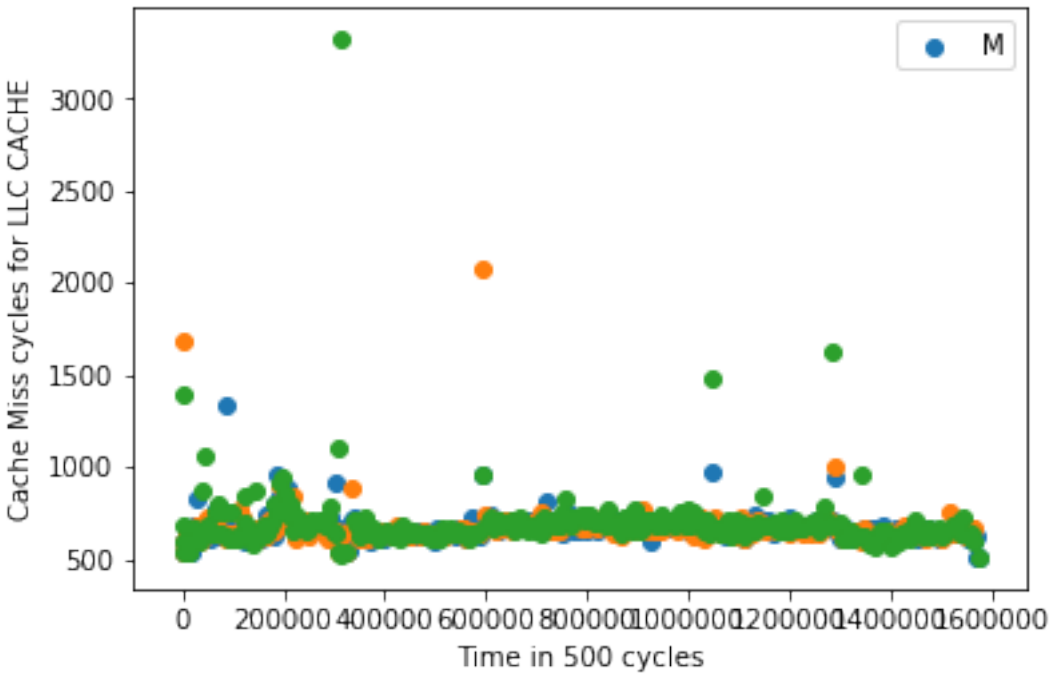
- We felt L1 Instruction cache is very noisy.
- To reduce noise : we did the following
  - We gave high priority for both attacker and victim
  - We resricted all the user processes which are mapped to same core (so that only victim and attacker are running in that core , to reduce noise)
- Even after following the above techniques for reducing the noise , still noise is present.
- As mentioned int Flush and Reload here also noise becase of
  Scheduling: As we know that different processes will be scheduled in same core, because of this victim and attacker are not scheduled at same time even though we are running them parallely
  Less Sets in L1 I Cache: as there are very less number of sets in L1 cache there is probablity of noise in L1 cache.

Accuray :

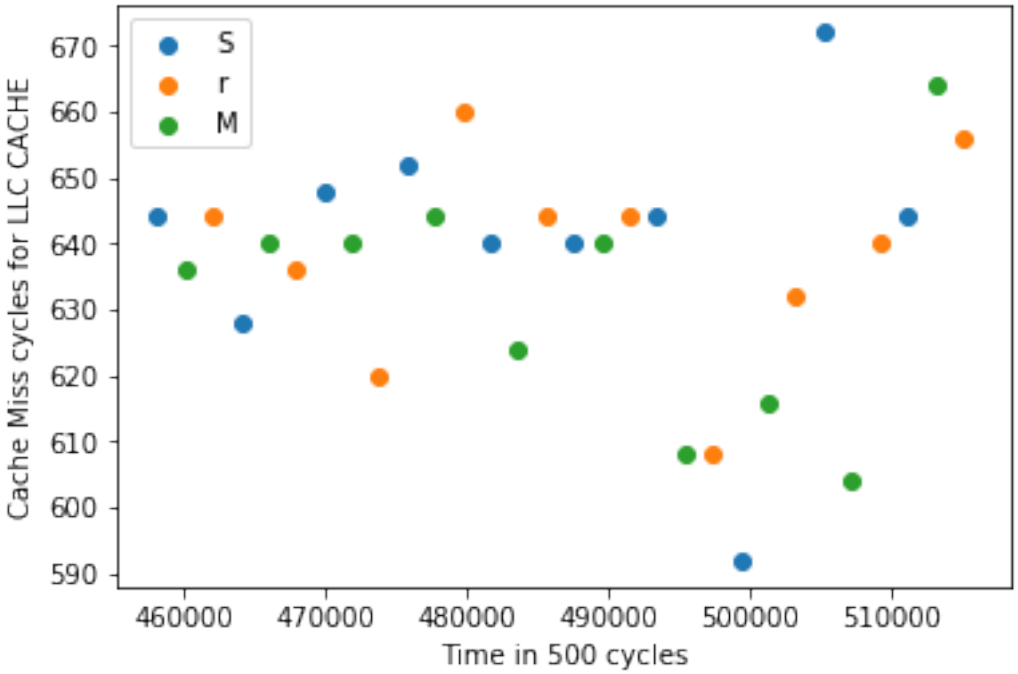
- We got 133 bits at a particular instance , whose edit distance 1915. It is about 6.5 % accuracy.

Prime and Probe on LLC Cache :

plot:



snapshot:



Insights:

- we created eviction sets
- we used those virtual addresses of square , reduce, multiply functions to find the corresponding it set addresses(possible because ASLR is OFF and Huge Page Table size is enabled).
- We moniored those sets by prime and probe , we found that without running victim program we got misses in those sets.