Appendix

Code Manual

The code consists of two python scripts - attacker.py and server.py and a c file flushreload.c. The flushreload.c file performs flush+reload on the 5 addresses mentioned in section 4.2. The server.py just accepts messages on a port, signs them and sends back. The attacker.py first generates random messages, sends them to server to get them signed, executes flushreload.c to perform side channel, store valid samples to finally solve the HNP and extract the private key.

Installation

The following packages need to be installed for the code to work -

1. Mastik - download from `https://cs.adelaide.edu.au/~yval/Mastik/`
2. apt install pkg-config
3. apt install libfplll-dev
4. Following python 2.7 packages -
   (a) Cython
   (b) cysignals
   (c) fpylll
   (d) cryptography

   It might be convenient to install these packages using Anaconda environment as we faced a lot of trouble installing fpylll manually. More information on how to use anaconda to manage python packages can be found here - `https://www.digitalocean.com/community/tutorials/how-to-install-the-anaconda-python-distribution-on-ubuntu-16-04`

Running the code

First make following changes to the code for it to be able to work on your machine -

1. Change the shebang in attacker.py and server.py to point to anaconda's python and system's python respectively (If packages are not installed via anaconda, change shebang in both files to point to system's python)
2. Change line 51 in attacker.py to /path/to/libcrypt.so.1.1/provided/by/us

3. You may need to change the THRESHOLD global variable in flushreload.c as it varies across machines

Perform the following steps to run the attack -

1. Generate keypair and set their privilege level -
   (a) sudo rm -f private.pem public.pem
   (b) openssl ecparam -genkey -name prime256v1 -noout -out private.pem
   (c) openssl req -new -x509 -key private.pem -out public.pem
   (d) chmod 666 public.pem
   (e) chmod 400 private.pem
   (f) sudo chown root:root private.pem
2. Build Flush+Reload binary -
   gcc -Imastik/src -Lmastik/src -o flushreload flushreload.c -lmastik
3. In one terminal do the following -
   (a) export LD_PRELOAD=$LD_PRELOAD:/path/to/libcrypt.so.1.1/provided /by/us
   This ensures that the cryptographic libraries find this particular libcrypt file in path before others and so uses this
   (b) sudo ./server.py &
4. In another terminal do -
   ./attacker.py numsamples knownbits
   Here, numsamples and knownbits denote the number of samples required to be collected by the attacker and the minimum known bits derived from a signature to consider it to formulate the HNP respectively.