Derive P*k* of NXP MIFARE Classic EV1 ECDSA Signature

19. Februar 2021

Needed:

- Proxmark3 with the latest firmware and client installed from https://github.com/RfidResearchGroup/proxmark3
- A git clone of https://github.com/RfidResearchGroup/proxmark3

(`r`, `s`, `v`) is the signature model used.

`r` can be read on PM3 with the command `hf mf rdbl 69 B 4b791bea7bcc`

`s` can be read on PM3 with the command `hf mf rdbl 70 B 4b791bea7bcc`

`v` is the UID of the card.

Example:

`r` is `19505576ED327D8F8870C86B1ED00898`

`s` is `BFEDFFF27CC82FC515BA2EEC26050873`

and `v`, which is the UID, is `BD2A4146`.

Utilize the `recover_pk.py` script from the root of the Proxmark3 repository, like below:

`python3 tools/recover_pk.py BD2A4146 19505576ED327D8F8870C86B1ED00898BFEDFFF27CC82FC515BA2EEC26050873`

You'll get an output like below:

```
linuxgemini@linuxgemini-fx:~/gitworkflow/proxmark3$ python3 tools/recover_pk.py
BD2A4146 19505576ED327D8F8870C86B1ED00898BFEDFFF27CC82FC515BA2EEC26050873

Assuming curve=secp128r1
========================
Assuming hash=None
Possible uncompressed Pk(s):
044f6d3f294dea5737f0f46ffee88a356eed95695dd7e0c27a591e6f6f65962baf
04e0a90de0f96a5c99d9bb6dae8252c86591945e886828bac51a20f47e71554a46
Assuming hash=md5
Possible uncompressed Pk(s):
045adce9091c63e52868fa983e518d8b0f8afafd0315a546f60c9c688a09115e51
0452e4e7c195f75e8b8bd0bdbead61ede72017a1c1f48607376c956d0e9c4764ad
Assuming hash=sha1
Possible uncompressed Pk(s):
0412710ae8d4b8383841e25258771d8ae2f15f3f5f0587ef148ea82c96e9bb2ceb
0479eda8ee440bf8a8cade01c52d22ac4c5fc77ca034188d1436190130565c44d2
Assuming hash=sha256
Possible uncompressed Pk(s):
04f6f1f263353789c6276ebf3fe64ea0043c1e8e2e7be204f65f1b133055b26d81
04284dd1719bb62d8368284320673c2fd21d04e815ec6e6f07ff009053f92ed6e5
Assuming hash=sha512
Possible uncompressed Pk(s):
0439cd2bbe5e6c81f388f2fc320dd097f014b33b941e708a50acbdacac8d56afeb
04dce974f5462e4ad2ddb3cfafb640a2ac32923552068c9f7b4e393bf5db397749

Assuming curve=secp128r2
========================
Assuming hash=None
Assuming hash=md5
Assuming hash=sha1
Assuming hash=sha256
Assuming hash=sha512
```

Compare the P*k* outputs with the test cases inside `recover_pk.py` to see which one will match, in this case its

`044F6D3F294DEA5737F0F46FFEE88A356EED95695DD7E0C27A591E6F6F65962BAF`

which is the P*k* of NXP MIFARE Classic EV1.
