QUEEN'S UNIVERSITY BELFAST

ESTD 1845

CSIT

CENTRE FOR SECURE INFORMATION TECHNOLOGIES

QUEEN'S UNIVERSITY BELFAST

CSIT CENTRE FOR SECURE INFORMATION TECHNOLOGIES

Plaintext: A Missing Feature for Enhancing the Power of Deep Learning in Side-Channel Analysis?

Breaking multiple layers of side-channel countermeasures

Anh-Tuan Hoang, Neil Hanley and Máire O'Neill

CHES 2020 14-18 September 2020

Outline

- Background and ASCAD database
- Attack model
- Plaintext feature in SCA
- Proposed CNNP models: hyperparameter and models
- Experimental conditions and reference models
- CNNP models evaluation
- Discussion

Side-channel Analysis (SCA)

- When an electronic device operates, it can leak data through side-channels, such as via power consumption, EM fields, timing
- Even though the cryptographic algorithm is secure in theory, secret information can be revealed from side-channel information
- SCA-based attacks like DPA and CPA are well known since 1996
- More recently, shown that machine learning can learn from side-channel information to reveal the secret key of a cryptographic device

Convolutional Neural Network (CNN)

- Can learn from unaligned data
- Includes a number of layers:
  - **Convolutional layers** based on a number of filters to detect features of the data
  - **Pooling layer** is used to reduce size of the parameters to be learned
  - **Fully connected layer** combines all previous features (nodes) together
  - **Dropout layer** is used to prevent over fitting by randomly removing a number of detected features (nodes)
- Activation functions
  - **Rectified Linear units** introduce non-linear computation into the output of a neuron
  - **Softmax** is used to handle the final classification

Evaluated AES implementation with SCA countermeasure (from ASCAD Database)

https://www.data.gouv.fr/en/datasets/ascad/

- Software implementation on 8-bit AVR ATMega 8515 microprocessor
- Two masks are used for
  - Plaintext

$$\overline{p_i} = p_i \oplus m_i$$

  - SBox

$$\overline{SBox(x)} = SBox\left(x \oplus m_{i,in}\right) \oplus m_{i,out}$$

ASCAD database

- Targeted the third sub-key, which is protected by two kinds of masking
- Fixed key dataset
  - Same key used for learning and testing
  - Trace length: 700 points
  - Training group: 50,000 traces
  - Testing group: 10,000 traces
- Variable key dataset
  - Random keys used in training data group and fixed key used in testing data group
  - Trace length: 1,400 points
  - Training group: 200,000 traces
  - Testing group: 100,000 traces
- Synchronized, desynchronized datasets are available

Attack model

- Attack on the output of the 3rd SBox in the 1st round of AES
- Classification uses the output value of SBox (256 classes)

$$SBox(p_2 \oplus k_2)$$

Plaintext feature in SCA

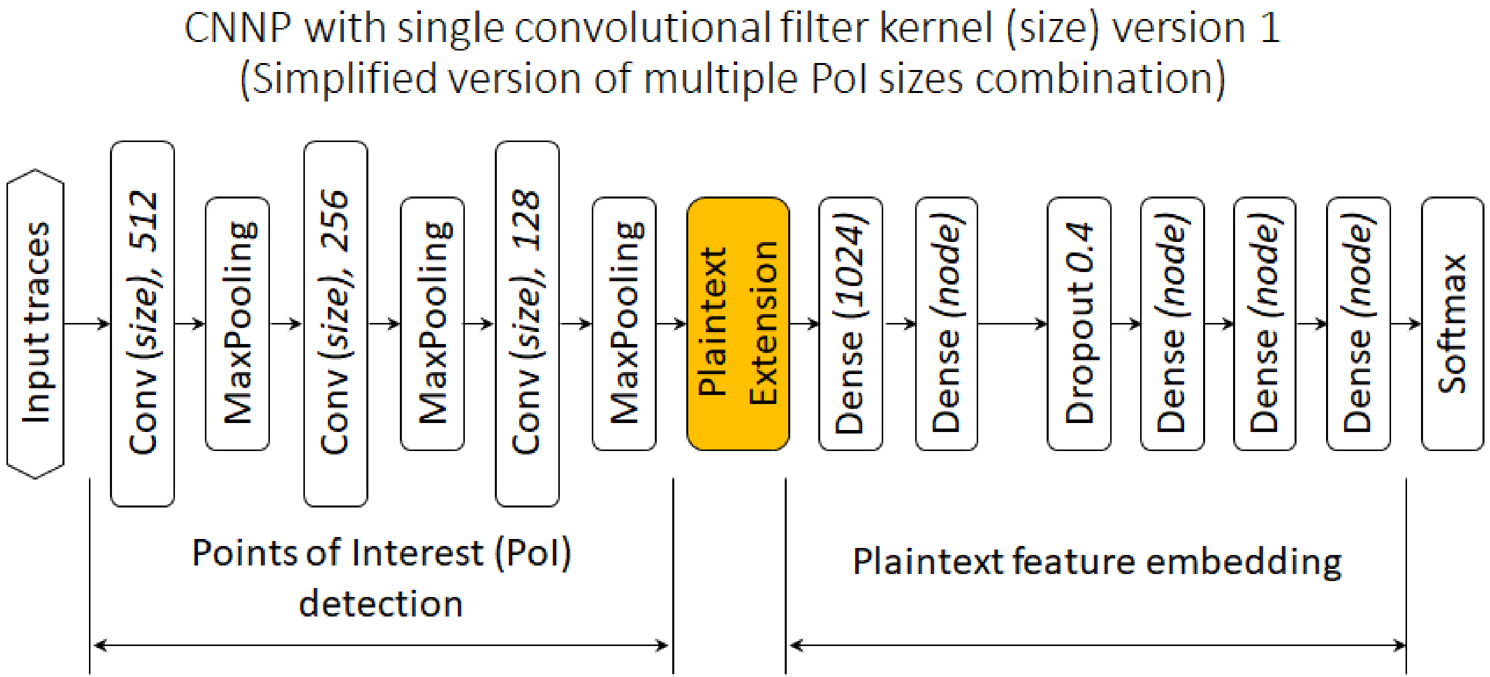
- Inputs that effect a power trace:
  - Plaintext (or ciphertext)
  - Masks
  - Key
- Providing plaintext or ciphertext reduces the number of unknown factors
- Plaintext feature is added using two methods: integer and one-hot encoding, where the feature is shown by a single number or a sparse vector

Proposed CNN model and hyperparameter selection

- Convolutional filter kernel sizes range from 3 to 19
- MaxPooling is used for local point of interest selection
- Convolutional layers have 64, 128, 256 and 512 filters
- Five fully connected layers of 1024 and 512 neurons each
- Activation function: ReLu

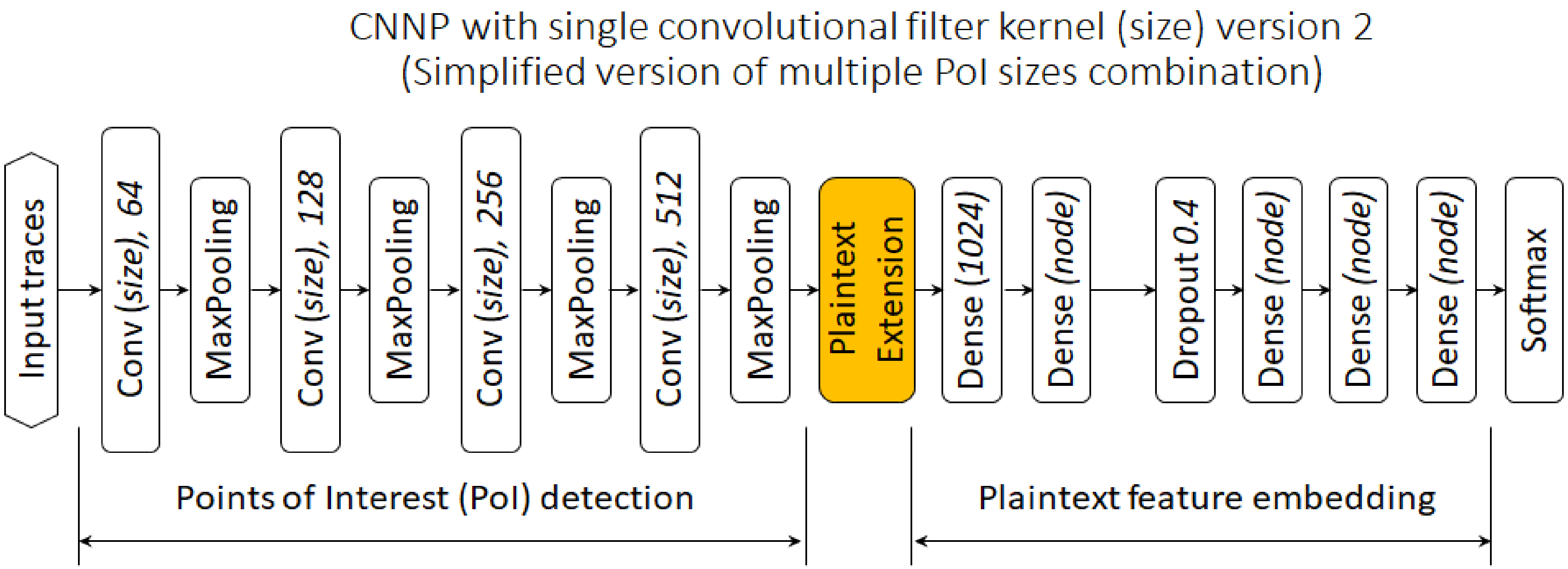
CNN with Plaintext extension (CNNP) – Model 1

- Three convolutional layers
- The number of convolutional filters reduces from 512 to 128
- Maxpooling is used for feature finding
- Finding features are extended with Plaintext



- Five fully-connected layers are used to compile the features extracted from the previous layers
- Over-fitting is prevented by using dropout

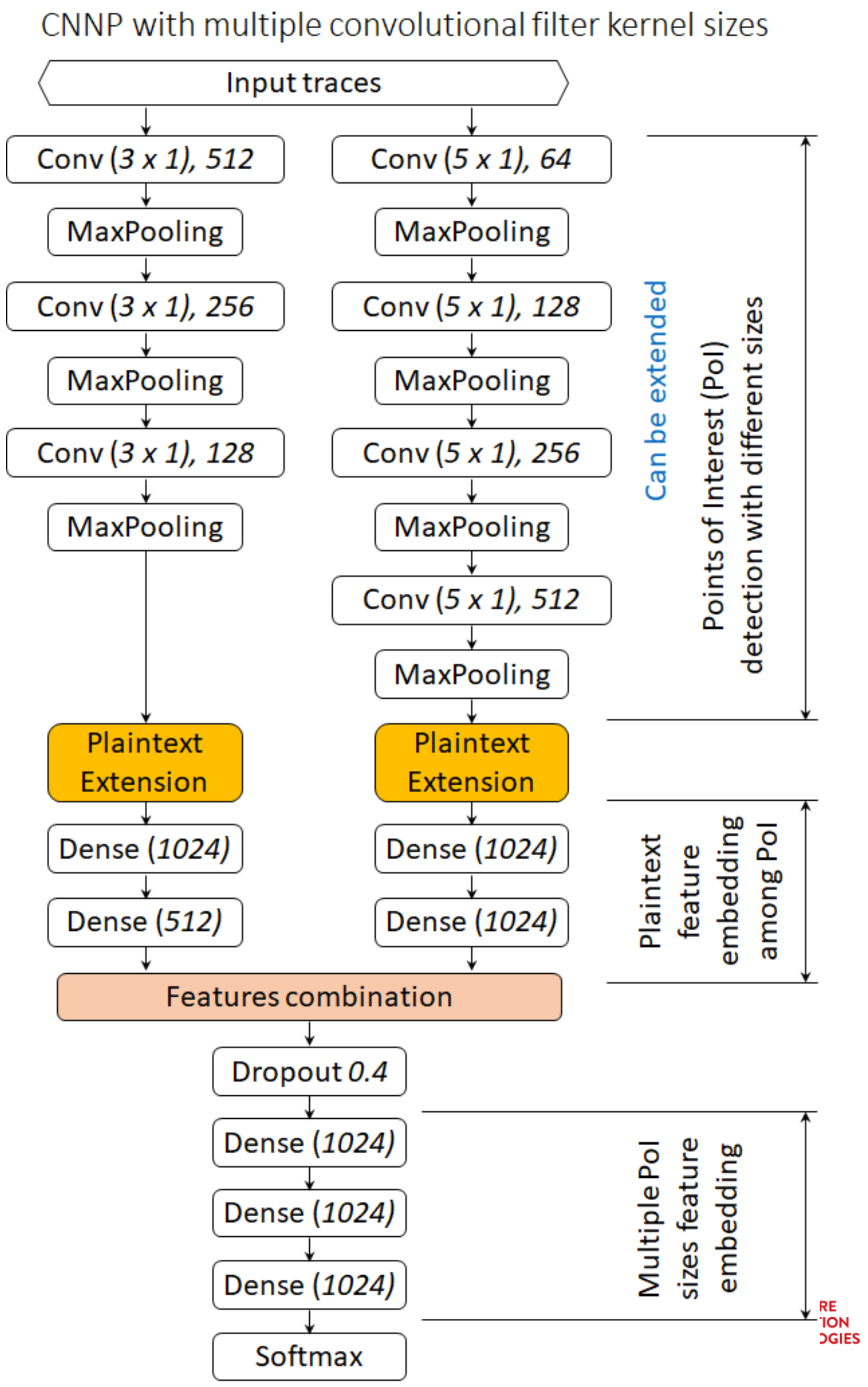
CNN with Plaintext extension (CNNP) – Model 2



- Four convolutional layers used
- The number of convolutional filters increases from 64 to 512
- Plaintext feature is extended by connecting to the detected features
- Five fully-connected layers are used

CNNP model extension

- Combination of CNNP models 1 and 2 using transfer learning
- Two fully-connected layers are used to compile the features extracted from each CNNP model before combination
- Three other fully-connected layers are used to combine the combination features
- Feature combination layer must be located after the fully-connected layers of the two CNNP sub-models



Attackers knowledge & experimental conditions

- Assumption about attacker:
  - Knows plaintext / ciphertext
  - Aware of SCA countermeasure but not aware of the detailed design and random mask value
  - Can profile keys on the implementation
- Hypothesis keys are ranked using Maximum likelihood score
- Training is performed on VMware hosted Ubuntu with access to virtual NVIDIA GRID M60-8Q and M40-4Q GPUs.

SCA reference models

We compare our profiling results with 4 publicly available models (ASCAD database)

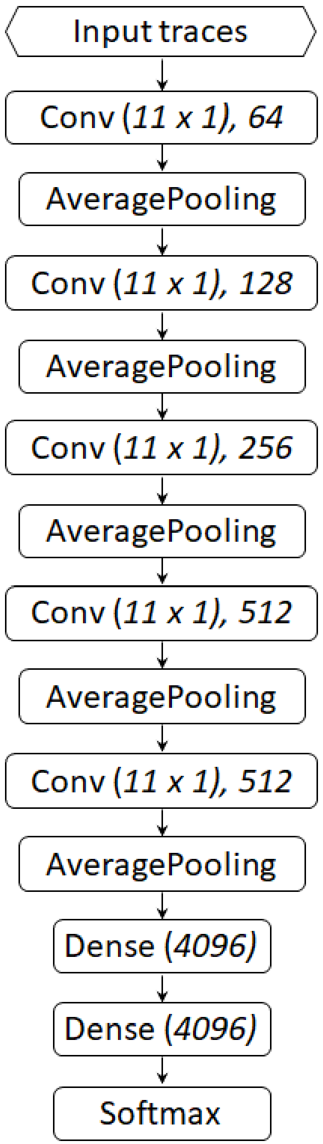
- Template attack
- Multilayer perceptron model with 5 hidden layers, 50 neurons each
- Multilayer perceptron model with 5 hidden layer - 700 neurons in first layer & 200 neurons in subsequent layers
- VGG-16 based CNN model

VGG16 Vs CNNP Models

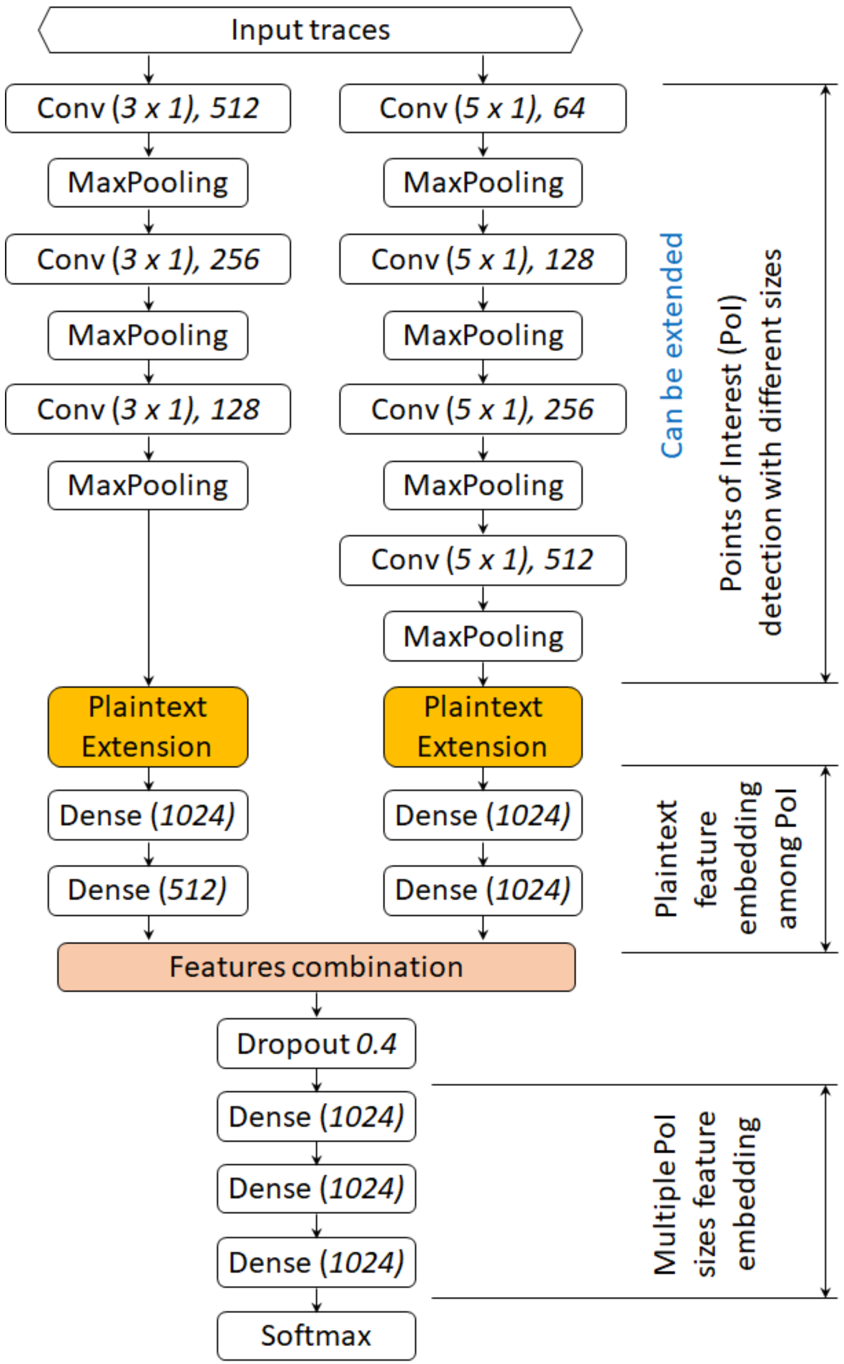
In comparison to the VGG-16 based model, the CNNP model:

- is deeper but narrower
- has less convolutional layers
- utilizes smaller convolutional filter kernel size
- uses MaxPooling instead of AveragePooling
- includes plaintext as an additional feature

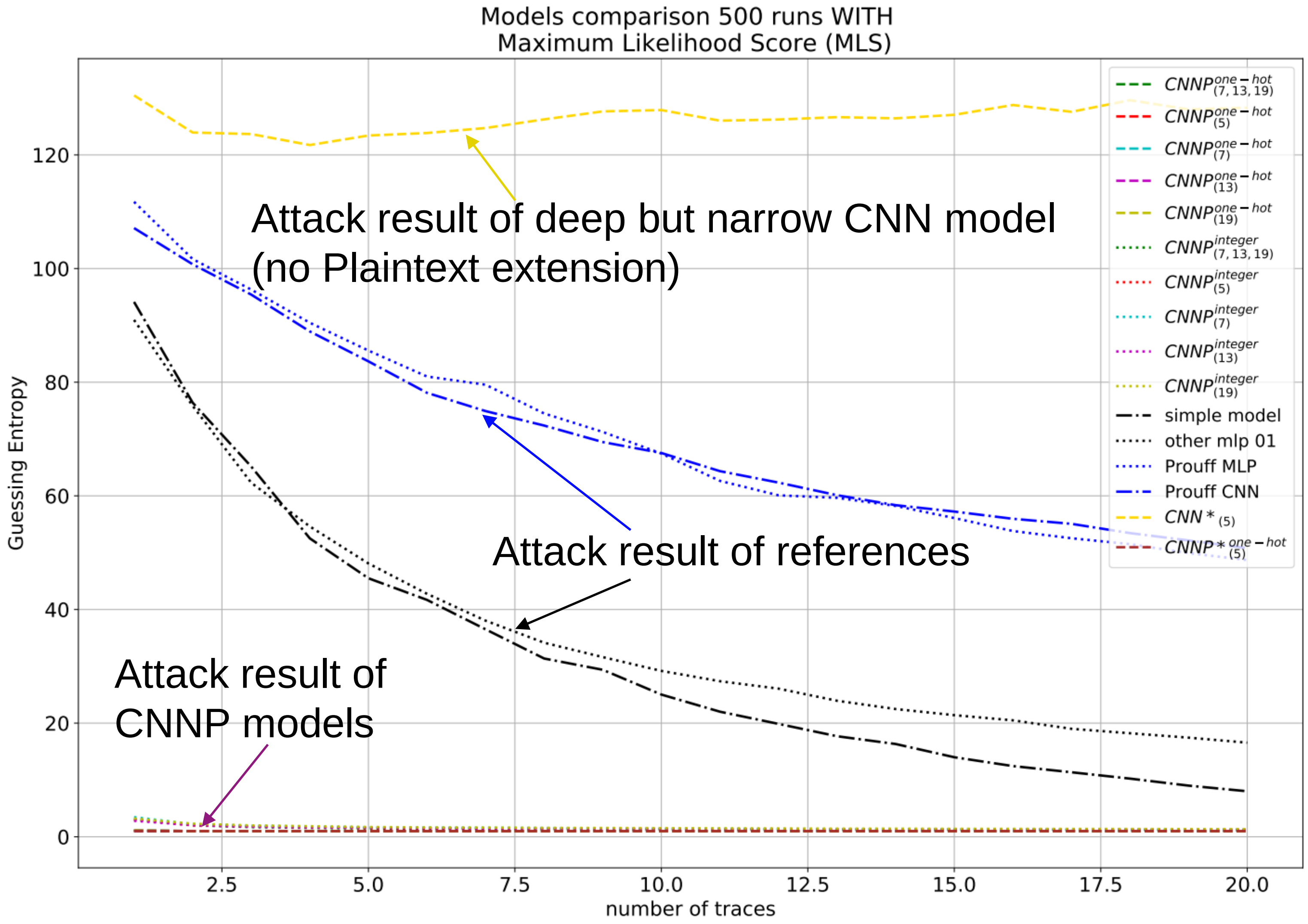
VGG-16 based CNN model



CNNP with multiple convolutional filter kernel sizes

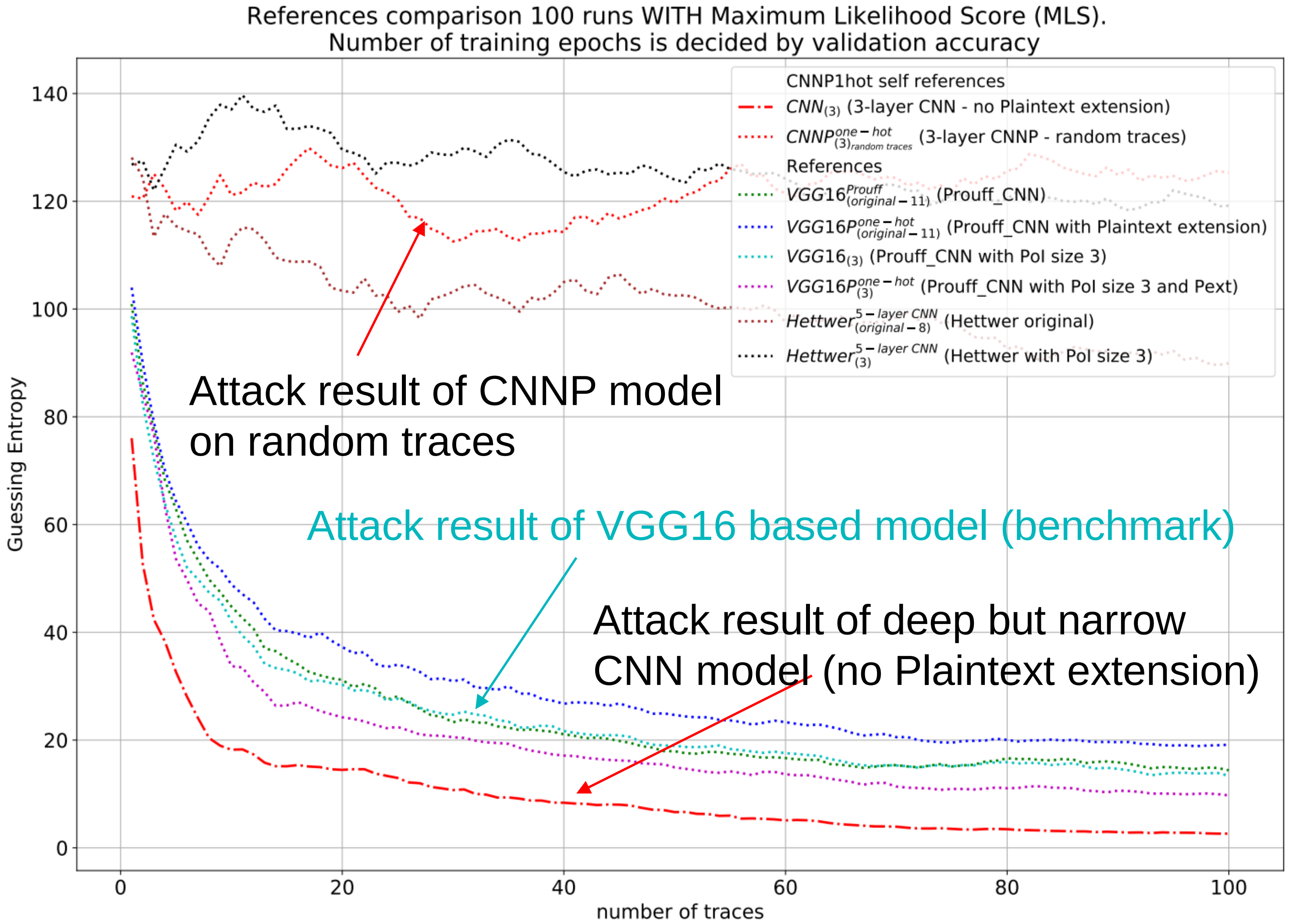


Evaluation of CNNP models on ASCAD fixed key dataset



- CNNP model can reveal the secret key within 2 traces
- CNNP models relies on the bijection $S[(.) \oplus K]$ to reveal K without using traces
- Plaintext feature encoded by one-hot encoding achieves better result than with integer encoding

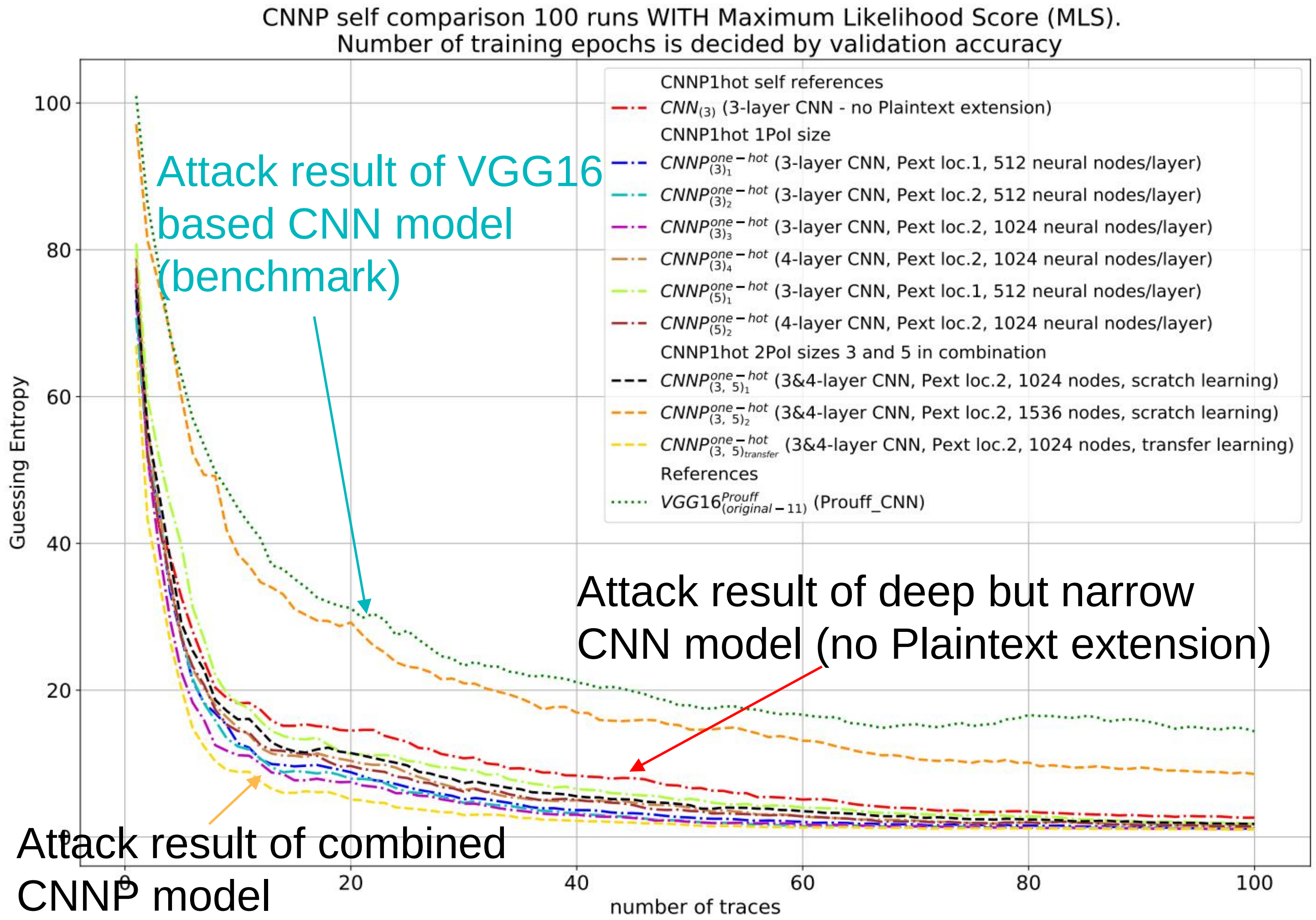
Evaluation of CNN models on ASCAD variable key synchronized dataset



- An additional reference model which refers to plaintext as a feature is included
- Proposed deep but narrow CNN model is better than all other models in revealing the secret key
- CNNP model on variable key relies on both plaintext and traces to learn

QUEEN'S UNIVERSITY BELFAST CSIT CENTRE FOR SECURE INFORMATION TECHNOLOGIES

Comparison of CNNP models on ASCAD synchronized dataset with variable key



- Both CNNP model 1 and 2 are better than VGG16 and and can achieve rank 3 and 5 for the 3rd subkey with 40 traces
- Smaller convolutional filter kernel size (e.g size 3) is more efficient than larger one (e.g. size 5)
- Combination of the 2 models with transfer learning achieves the best result

Discussion

- Effect of convolutional layers and filter sizes
  - Help to find the feature regardless of misalignment in the traces.
  - Small convolutional kernel size works better than larger kernel sizes
- Effect of Plaintext feature extension and location
  - Plaintext feature extension reduces the number of unknown factors that contribute to features in the traces
  - Location of plaintext feature has less effect on the result
- Effect of network structure
  - Deep but narrow network shows better attacking result than wide but shallow ones

Thank you

CENTRE FOR SECURE INFORMATION TECHNOLOGIES