VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2016 Stanislav Vaněk

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

METODY LOKALIZACE ZAJÍMAVÝCH BODŮ U PROUDOVÉ ANALÝZY
LOCALIZATION OF INTERESTING POINTS IN POWER ANALYSIS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR
Stanislav Vaněk

VEDOUCÍ PRÁCE
SUPERVISOR
Ing. Zdeněk Martinásek, Ph.D.

BRNO 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Stanislav Vaněk ***ID:*** 164431
Ročník: 3 ***Akademický rok:*** 2015/16

NÁZEV TÉMATU:

Metody lokalizace zajímavých bodů u proudové analýzy

POKYNY PRO VYPRACOVÁNÍ:

V rámci bakalářské práce nastudujte problematiku proudové analýzy, zaměřte se na metody lokalizace zajímavých bodů (Normalized InterClass Variance (NICV), Sum Of Squared pairwise Differences (SOSD), Sum Of Squared pairwise Tdifferences (SOST), Principal Components Analysis (PCA) nebo Pearson Correlation). Vytvořte přehledný rozbor současného stavu problematiky. S proudovými průběhy z http://www.dpacontest.org (verze 4.2) realizujte porovnání jednotlivých metod. Dosažené výsledky přehledně zpracujte. Jednotlivé metody (minimálně dvě) implementujte v prostředí Matlab a porovnejte s CPA.

DOPORUČENÁ LITERATURA:

[1] MANGARD, Stefan a OSWALD, Elisabeth a POPP, Thomas: Power Analysis Attacks: Revealing the Secrets of Smart Cards (Advances in Information Security). Secaucus, NJ, USA:Springer-Verlag New York, Inc., 2007, ISBN 0387308571.

[2] BHASIN, Shilvam a DANGER, Jean-Luc a GULILLEY, Sylvain a NAJM, ZakariaBhasin: Side-channel leakage and trace compression using normalized inter-class variance. In Proceedings of the Third Workshop on Hardware and Architectural Support for Security and Privacy (p. 7). ACM.

Termín zadání: 1.2.2016 ***Termín odevzdání:*** 1.6.2016

Vedoucí práce: Ing. Zdeněk Martinásek, Ph.D.
Konzultant bakalářské práce:

doc. Ing. Jiří Mišurec, CSc., *předseda oborové rady*

ABSTRAKT

V současnosti je zapotřebí klást na kryptografická zařízení stále větší nároky kvůli narůstajícímu počtu zneužití. Z tohoto důvodu je nutné znát principy zabezpečení a jejich nedostatky. Tato práce se zabývá metodou lokalizace zajímavých bodů u proudové analýzy. Cílem práce je provést rozbor těchto metod a jejich následná implementace.

KLÍČOVÁ SLOVA

SPA, DPA, proudový model, rozptyl, NICV, kovariance, Sbox, korelační koeficient, CPA, PCA, hlavní komponenta, vlastní vektor, vlastní číslo

ABSTRACT

Nowadays there are very high demands on security of cryptographic devices due to the increasing number of exploitation. Because of these reasons it is necesary to know the principles of security and their flaws. This thesis deals with the method of localization of interesting markers in current analysis. The aim of the thesis is to analyze this methods and their potential implementation in practise.

KEYWORDS

SPA, DPA, current model, variance, NICV, covariance, Sbox, correlation coefficient, CPA, PCA, principal component, eigenvector, eigenvalue

VANĚK, S. *Metody lokalizace zajímavých bodů u proudové analýzy.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2016. 33 s. Vedoucí semestrální práce Ing. Zdeněk Martinásek, Ph.D..

Prohlášení

Prohlašuji, že svou semestrální práci na téma „Metoda lokalizace zajímavých bodů u proudové analýzy“ jsem vypracoval samostatně pod vedením vedoucího semestrální práce, s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce.

Zároveň jako autor semestrální práce prohlašuji, že v souvislosti s vytvořením této semestrální práce nebyla porušena autorská práva třetích osob, zejména nebylo nedovoleným způsobem zasaženo do cizích autorských práv osobnostních, nebo majetkových. Jsem si plně vědom následků porušení ustanovení §11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku
č. 40/2009 Sb.

V brně …….. ……..

podpis autora

Poděkování

Tímto bych chtěl poděkovat vedoucímu práce panu Ing. Zdeňku Martináskovi, Ph.D. za odborné vedení, konzultace a užitečné rady týkající se dané problematiky.

V brně …….. ……..

podpis autora

SIX research centre
sensor, information and communication systems

Faculty of Electrical Engineering
and Communication

Brno University of Technology
Technicka 12, CZ-61600 Brno, Czechia

http://www.six.feec.vutbr.cz

Výzkum popsaný v této bakalářské práci byl realizovaný v laboratořích podpořených projektem Centrum senzorických, informačních a komunikačních systémů (SIX); registrační číslo CZ.1.05/2.1.00/03.0072, operačního programu Výzkum a vývoj pro inovace.

OBSAH

SEZNAM OBRÁZKŮ

ÚVOD

V poslední době jsou útoky postranními kanály velmi populární. Pro bezpečnost kryptografických modulů představují významná rizika. Existuje řada útoků, které lze aplikovat na prolomení šifrovacích algoritmů, jako jsou např. RSA, DES, AES. K realizování útoků postranními kanály jsou nezbytnou součástí metody proudové analýzy, pro představu například jednoduchá proudová analýza (SPA), diferenciální proudová analýza (DPA) aj. Výše uvedené metody jsou v této práci podrobně popsány, a to konkrétně v první kapitole. Druhá kapitola se zabývá základními modely proudové spotřeby. Ty jsou používány u proudové analýzy kryptografických zařízení. Třetí kapitola obsahuje samotný rozbor metod lokalizace zajímavých bodů, a to sice metody DPA založené na korelačním koeficientu (CPA), metody založené na analýze rozptylu (NICV) a metody předzpracování dat (PCA).

1 METODY PROUDOVÉ ANALÝZY

V následující kapitole jsou popsány dvě metody proudové analýzy, které jsou používány pro odhalení tajného klíče v kryptografickém zařízení.

1.1 Jednoduchá proudová analýza

Simple Power Analysis (SPA), neboli také jednoduchá proudová analýza, pracuje na principu přímého pozorování odběru proudu kryptografického zařízení, kdy se útočník snaží pomocí tohoto pozorování zjistit, jakou činnost provádí zařízení v určitém čase. Pro útočníka je ovšem nutná dokonalá znalost kryptografického zařízení a také implementace šifrovacího algoritmu. Pokud má útočník k dispozici z celé proudové spotřeby pouze jeden průběh, jedná se o analýzu jednoho proudového průběhu, nicméně tento případ se vyskytuje spíše ojediněle. Většinou útočník disponuje hned několika proudovými průběhy, které jsou změřeny buď pro stejný otevřený text, nebo pro více různých textů. Pro jeden otevřený text je několik proudových průběhů značnou výhodou, díky schopnosti redukce šumu. Útok za pomocí SPA může být prováděn například na šifrovacím algoritmu DES, kde jsou dobře pozorovatelné rozdíly proudové spotřeby při vykonávání různých operací, konkrétně posunu a permutace [16]. Stejně tak může být prováděn útok na šifrovací algoritmus RSA či AES. Podrobný popis problematiky zaměřující se na SPA definoval Kocher. [8]

1.2 Diferenciální proudová analýza

Při použití diferenciální proudové analýzy (DPA) na rozdíl od SPA není potřebná dokonalá znalost kryptografického zařízení, dokonce pomocí DPA mohou být odhaleny klíče i v případě, kdy zaznamenané proudové průběhy obsahují značné množství šumu. Na druhou stranu oproti SPA potřebuje útočník velké množství proudových průběhů zachycených v momentě šifrování nebo dešifrování vstupních dat. DPA zkoumá závislost proudové spotřeby na zpracovávaných datech. DPA útoky jsou prováděny v pěti krocích [4]:

Krok první: Zvolení mezivýsledku algoritmu

Kryptografickým zařízením je vykonáván šifrovací algoritmus, jehož mezivýsledek je zvolen. Je nezbytné, aby byl tento mezivýsledek funkcí $f(d, k)$, kde d představují známá vstupní data (tzv. PlainText nebo CipherText, neboli otevřený či šifrovaný text) a k je malou částí (např. první bajt) šifrovacího klíče. [11] [16]

Krok druhý: Měření průběhů proudové spotřeby

Měření proudové spotřeby kryptografického zařízení vykonávající šifrování nebo dešifrování různých bloků dat D je druhým krokem DPA útoku. Útočník potřebuje pro veškeré operace šifrování nebo dešifrování znát hodnoty zpracovávaných dat d, která se podílí na výpočtu mezivýsledku, který je zvolen v kroku prvním. [11] [16]

Hodnoty známých dat tvoří vektor $\boldsymbol{d} = (d_1, \dots, d_D)'$, kde d_i je hodnota i-tého bloku vstupních dat, který je zpracováván. Útočník si při vykonávání těchto operací značí proudovou spotřebu kryptografického zařízení. Bloky dat d_i korespondují s průběhy proudové spotřeby, které můžeme označit jako $t'_i = (t_{i,1}, \dots, t_{i,T})$, kde T je délka naměřené proudové spotřeby. Jelikož je útočníkem měřena proudová spotřeba pro každý blok zpracovávaných dat D, mohou být průběhy zapsány jako matice $\boldsymbol{T}$, která má velikost $D \times T$. U DPA útoku je rozhodující, aby naměřené průběhy proudové spotřeby byly přesně zarovnané, a to tak, že pro hodnoty sloupce t_j matice $\boldsymbol{T}$ musí odpovídat stejné operaci. Správně zarovnaná data zajišťuje správná synchronizace měřícího zařízení. [11] [16]

Krok třetí: Sestrojení matice hypotéz mezivýsledků

Třetím krokem útoku je určit hypotetické mezivýsledky pro všechny možné hodnoty klíče k. Hodnoty klíče k můžeme zapsat jako vektor $\boldsymbol{k} = (k_1, \dots, k_K)$, kde K je celkový počet všech možných klíčů. Jednotlivé prvky vektoru k bývají označovány jako hypotézy (odhady) klíče. Pro všechny odhady klíče K a pro všechny šifrovací operace D je útočník schopný z vektorů hypotéz všech klíčů a vektorů veřejných dat $\boldsymbol{d}$ spočítat hodnotu mezivýsledku $f(d, k)$. Výsledek tvoří matice $\boldsymbol{V}$, která má rozměry $D \times K$ daná podle následujícího vztahu [11] [16]:

$$v_{i,j} = f(d_i, k_j)\ i = 1, \dots, D\ j = 1, \dots, K \tag{1.2}$$

Mezivýsledky spočítány podle hypotéz klíče k_j jsou obsaženy ve sloupci j matice $\boldsymbol{V}$. Právě jeden sloupec matice $\boldsymbol{V}$ obsahuje ty mezivýsledky, které kryptografické zařízení vypočítalo během procesu šifrování a dešifrování. Index tohoto sloupce je označen c_k. Potom hledaný klíč má tvar k_{ck}. DPA útok má za úkol nalézt právě ten sloupec, který byl zpracováván při šifrování a dešifrování. Tudíž najít klíč k_{ck}. [11] [16]

Krok čtvrtý: Přiřazení mezivýsledků na hodnoty proudové spotřeby

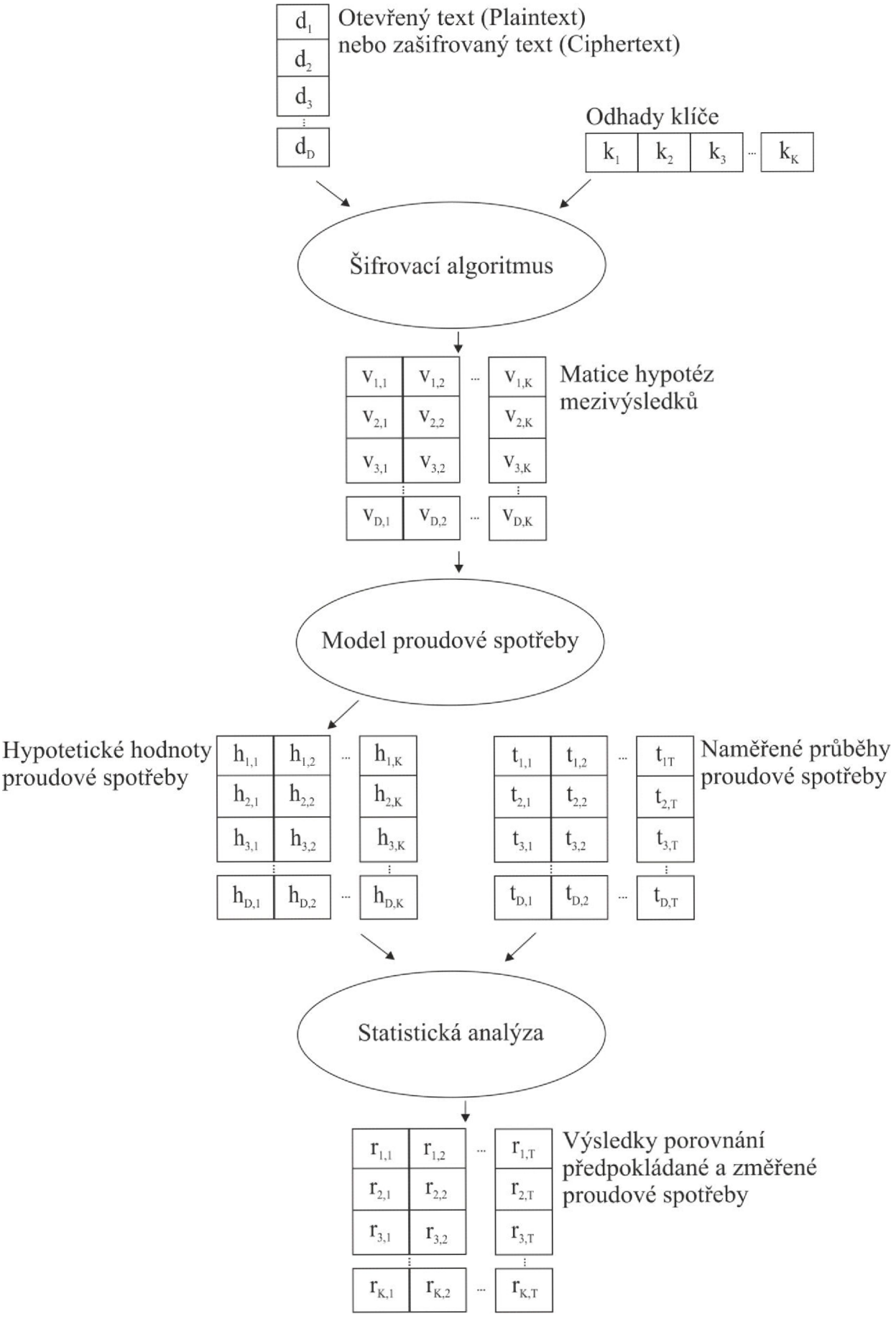
Ve čtvrtém kroku DPA útoku je namapování matice $\boldsymbol{V}$, ve které jsou obsaženy hypotézy mezivýsledků, na matici $\boldsymbol{H}$ reprezentující hypotetické hodnoty proudové spotřeby. Tady je využita simulace proudové spotřeby kryptografického zařízení. Zvolený model spotřeby přiřadí každému očekávanému mezivýsledku $v_{i,j}$ hypotetickou hodnotu proudové spotřeby $h_{i,j}$. [11] [16]

Pokud má útočník velké znalosti o analyzovaném zařízení, je schopný vytvořit lepší simulaci proudové spotřeby a DPA bude efektivnější. Co se týče využívaných modelů spotřeby, tak mezi nejpopulárnější patří model Hammingovy váhy a model Hammingovy vzdálenosti, které budou popsány ve druhé kapitole. [11] [16]

Krok pátý: Porovnání naměřených hodnot proudové spotřeby s předpokládanými hodnotami

V tomto kroku jsou útočníkem porovnávány hypotetické hodnoty proudové spotřeby, které jsou závislé na odhadu klíče (hodnoty sloupce h_i matice $\boldsymbol{H}$), s hodnotami změřených průběhů proudové spotřeby (hodnoty sloupce t_j v matici $\boldsymbol{T}$). Výsledek tvoří matice $\boldsymbol{R}$ velikosti $K \times T$. [11] [16]

Všechny prvky $r_{i,j}$ matice $\boldsymbol{R}$ představují výsledky porovnání mezi sloupci h_i a t_j. Porovnání samotné může být realizováno pomocí tzv. metod lokalizace zajímavých bodů u proudové analýzy, některé z nich budou podrobně probrány ve třetí kapitole. Proudová spotřeba kryptografického zařízení při šifrovacím algoritmu pro různá data je vyjádřena naměřenými průběhy. Součástí tohoto algoritmu je i zvolený výsledek z prvního kroku. Tudíž během každého šifrování nebo dešifrování vstupních dat pracuje zařízení s mezivýsledky v_{ck}. Naměřené průběhy jsou tedy v určitých místech závislé na hodnotě mezivýsledků. Toto určité místo označíme ct. Z toho vyplývá, že se ve sloupci t_{ct} nacházejí hodnoty proudové spotřeby, které jsou závislé na mezivýsledku v_{ck}. Na základě právě hodnot v_{ck} byly útočníkem simulovány předpokládané hodnoty proudové spotřeby h_{ck}. Z tohoto důvodu jsou sloupce h_{ck} a t_{ct} na sobě velmi závislé. K největší hodnotě v matici $\boldsymbol{R}$ vedou právě tyto dva sloupce. To znamená, že hodnota $r_{ck,ct}$ je největší hodnota v matici $\boldsymbol{R}$. Jelikož na sobě jiné sloupce matic $\boldsymbol{H}$ a $\boldsymbol{T}$ nejsou tolik závislé, tak ostatní prvky matice $\boldsymbol{R}$ nebudou dosahovat tak vysokých hodnot. Jestliže útočník nalezne největší hodnotu v matici $\boldsymbol{R}$, pak nalezl správný klíč k_{ck}. Výsledkem DPA útoku jsou tedy indexy největší hodnoty v matici $\boldsymbol{R}$. V praxi může nastat případ, kdy hodnoty matice $\boldsymbol{R}$ budou nabývat skoro totožných hodnot. To je způsobeno nedostatkem změřených proudových průběhů ke stanovení závislosti mezi sloupcem matice $\boldsymbol{H}$ a sloupcem matice $\boldsymbol{T}$. S větším počtem změřených průběhů lze lépe charakterizovat závislost mezi těmito sloupci. [11] [16]



Obr. 1.2: Blokový diagram znázorňující kroky 3 až 5 DPA útoku. [4]

2 ZÁKLADNÍ MODELY PROUDOVÉ SPOTŘEBY

U diferenciální proudové analýzy je nezbytnou součástí přiřazování mezivýsledků na hodnoty proudové spotřeby. Tento postup je nazýván simulací proudové spotřeby kryptografického modulu. Při DPA útoku nehrají velkou roli samotné absolutní hodnoty proudové spotřeby, nýbrž relativní rozdíly mezi napodobenými hodnotami proudové spotřeby. Útočník v mnoha případech nemá k dispozici podrobnější znalosti o napadaném kryptografickém zařízení, proto jsou simulace proudové spotřeby poměrně jednoduché. Následující část obsahuje dva modely proudové spotřeby, a to model Hammingovy váhy a model Hammingovy vzdálenosti. [16]

2.1 Model Hammingovy váhy

Za předpokladu, že útočník nedisponuje žádnými znalostmi o vnitřní struktuře napadaného zařízení nebo nezná aktuálně zpracovávaná data, bude použit model Hammingovy váhy (Hamming weight). Předpokládá se, že datová spotřeba je přímo úměrná počtu nastavených nenulových bitů ve zpracovávaných datech [11]. V případě, že budou všechny datové bity nastaveny do logické 0, bude proudová spotřeba modelu nižší, než za předpokladu, že jsou všechny datové bity nastaveny do logické 1.

Tento model by však měl být užíván jen v omezených podmínkách z důvodu nedokonalého popisu proudové spotřeby modulu. [16][11]

2.2 Model Hammingovy vzdálenosti

U modelu Hammingovy vzdálenosti (Hamming distance) útočník očekává, že je proudová spotřeba úměrná počtu změněných míst ve zpracovaných datech [11]. Hlavním cílem tohoto modelu je určení konkrétní počet přechodů mezi logickými hodnotami z 0 na 1 a z 1 na 0 za určitý časový interval. Tento výsledný počet přechodů se pak používá k popisu proudové spotřeby obvodu. [16]

Mezi dvěma hodnotami je Hammingova vzdálenost za pomocí Hammingovy váhy dána vztahem: $HD(v_0, v_1) = HW(v_0 \oplus v_1)$. V mnoha případech je model Hammingovy vzdálenosti užíván pro popis proudové spotřeby registrů, či datových sběrnic. [16]

3 METODY LOKALIZACE ZAJÍMAVÝCH BODŮ U PROUDOVÉ ANALÝZY

Tato kapitola se zabývá matematickými metodami, díky kterým lze určit závislost mezi hypotetickou a skutečnou hodnotou proudové spotřeby, a to sice metodou DPA založenou na korelačním koeficientu (Correlation Power Analysis – CPA), dále metodou založenou na analýze rozptylu NICV (Normalized Inter-Class Variance) a v poslední řadě analýzou hlavních komponent (Principal Component Analysis – PCA).

3.1 Metoda založená na korelačním koeficientu (CPA)

Korelační koeficient je nejčastější způsob určení lineární závislosti mezi dvěma náhodnými proměnnými X a Y. Proto je také vynikající volbou v DPA útoku. Je definován pomocí kovariance vztahem [11]:

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sqrt{\sigma^2(X)\cdot\sigma^2(Y)}}, \tag{3.1.1}$$

kde $Cov(X,Y)$ značí tzv. kovarianci, což je střední hodnota součinu odchylek obou náhodných proměnných X a Y od jejich středních hodnot. Jinými slovy kovariance značí míru vzájemné vazby mezi dvěma náhodnými proměnnými X a Y. $\sigma^2(X)$ je rozptyl náhodné veličiny X a $\sigma^2(Y)$ zase rozptyl náhodné veličiny Y. Pro diskrétní náhodnou veličinu je rozptyl, neboli také variance je blíže definován vztahem:

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})^2}{(n-1)}, \tag{3.1.2}$$

kde n značí počet prvků (velikost populace).

Kovariance, jak již bylo řečeno, je definována jako střední hodnota součinu odchylek dvou náhodných proměnných X a Y od jejich středních hodnot:

$$Cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \overline{X})\cdot(Y_i - \overline{Y})}{(n-1)} \tag{3.1.5}$$

Korelační koeficient $\rho(X,Y)$ může nabývat hodnot $-1 \leq \rho \leq 1$. Za předpokladu, že korelační koeficient má hodnotu právě -1, můžeme říct, že se jedná o nepřímou závislost (změna v jedné grupě je doprovázena opačnou změnou v grupě druhé). [11]

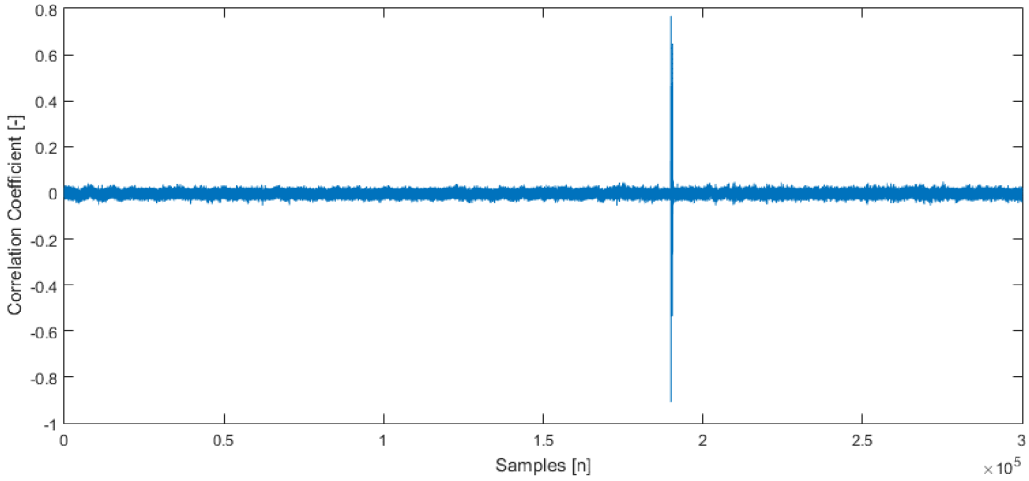
Pokud korelační koeficient nabývá hodnoty 0, znamená to, že mezi hodnotami grup není žádná lineární závislost. Jestliže má korelační koeficient hodnotu 1, tak byla nalezena přímá závislost mezi hodnotami obou grup. Za předpokladu, že jsou X a Y kvantitativní náhodné veličiny se společným dvourozměrným normálovým rozdělením, lze korelační koeficient pro konkrétní hodnoty $(x_1, y_1), \ldots, (x_n, y_n)$ [11] vyjádřit jako:

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})\cdot(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \cdot \sum_{i=1}^{n}(y_i - \overline{y})^2}}. \tag{3.1.6}$$

Při DPA útoku je korelační koeficient využíván k určení lineární závislosti mezi sloupci $\boldsymbol{h_i}$ a $\boldsymbol{t_j}$, kde $i = 1, \dots, K$ a $j = 1, \dots, T$ [24]. Výsledkem tohoto vztahu je matice $\boldsymbol{R}$, ve které jsou již obsaženy korelační koeficienty:

$$r_{i,j} = \frac{\sum_{d=1}^{D}(h_{d,i} - \overline{h_i}) \cdot (t_{d,j} - \overline{t_j})}{\sqrt{\sum_{d=1}^{D}(h_{d,i} - \overline{h_i})^2 \cdot \sum_{d=1}^{D}(t_{d,j} - \overline{t_j})^2}}, \tag{3.1.7}$$

kde $\overline{h_i}$ a $\overline{t_j}$ vyjadřují střední hodnoty sloupců $\boldsymbol{h_i}$ a $\boldsymbol{t_j}$.



Obr. 3.1: Matice korelačních koeficientů

Obr. 3.1 ukazuje výpis matice korelačních koeficientů $\boldsymbol{R}$ u DPA útoku prováděném na 300 000 průběhů proudové spotřeby, které jsou zašifrovány algoritmem AES (Advanced Encryption Standard). Je zřejmé, v jakém místě je lineární závislost největší, a tedy, kde se nachází citlivá informace, kterou se snaží útočník zjistit.

3.2 Normalized Inter-Class Variance (NICV)

Tato metoda umožňuje útočníkovi detekovat jednorozměrné úniky, tzv. first-order-attack. Oproti jiným metodám má užití NICV výhodu v tom, že používá stejnou sadu průběhů, které mají být analyzovány, tudíž není potřeba profilovací fáze (klon zařízení). Počítá s veřejnými parametry jako je otevřený text (Plaintext) a zašifrovaný text (Ciphertext). Nezávisí na zvoleném proudovém modelu. Ovšem je nutno poznamenat, že metoda NICV samotná není nástroj pro vykonání proudové analýzy, pouze usnadňuje a urychluje proudovou analýzu (např. CPA). [22] [17]

Označme si jeden bajt otevřeného nebo zašifrovaného textu označen X a proudovou spotřebu měřenou útočníkem $\mathrm{Y} \in \mathbb{R}$. Obě náhodné proměnné jsou veřejně známé. Pak pro veškeré uniklé funkce L z dané proudové spotřeby známe hodnotu x z oboru X: [12]

$$\rho^2[L(X);Y] = \rho^2\left[L(X);\ \mathbb{E}[Y|X|]\right] \times \rho^2[\mathbb{E}[Y|X|];\ \mathrm{Y}]\ , \qquad (3.2.1)$$

$$0 \leq \rho^2\left[L(X);\ \mathbb{E}[Y|X|]\right] \leq 1$$

$$\text{kde} \quad \rho^2[\mathbb{E}[Y|X|];\ \mathrm{Y}] = \frac{\mathrm{Var}\,\mathbb{E}[Y|X|]}{\mathrm{Var}\,[Y]}\ , \qquad (3.2.2)$$

což je označováno jako NICV. Jedná se o analýzu rozptylu tzv. ANOVA (Analysis Of Variance) [7].

Kombinací rovnic (3.2.1) a (3.2.2) vznikne rovnice platná pro všechny predikce funkce $\mathrm{L}: \mathbb{F}_2^8 \rightarrow \mathbb{R}$:

$$0 \leq \rho^2[L(X);Y] \leq \frac{\mathrm{Var}\,\mathbb{E}[Y|X|]}{\mathrm{Var}\,[Y]} = \mathrm{NICV} \leq 1, \qquad (3.2.3)$$

kde $\mathrm{Var}\,\mathbb{E}[Y|X|]$ odhadovaný rozptyl a $\mathrm{Var}\,[Y]$ označuje skutečný rozptyl.

Z této rovnice vyplývá, že NICV je maximum (obálka) všech možných korelací vypočitatelných z X a jim odpovídajících Y.

NICV, stejně jako CPA, dosahuje konstantní hodnotu asymptoticky, jakmile měřící soubor dosáhl představovanou velikost vzorku.

V praxi ovšem hodnota CPA nikdy nedosáhne hodnoty NICV vlivem šumu a jiných nedokonalostí. Důvodem rozdílu může být:

- útočník zná přesný odhad funkce, ale nezná (což je obvyklé) skutečný klíč. Pro představu předpokládejme, že průběhy lze zapsat jako: $Y = w_H\left(S(X \oplus k^*)\right) + N$, kde $k^* \in \mathbb{F}_2^8$ je správný klíč, $S: \mathbb{F}_2^8 \rightarrow \mathbb{F}_2^8$ je substituční box (Sbox), w_H je Hammingova váha a N je měřený šum. V tomto případě je optimální odhad funkce: $L(x) = \mathbb{E}[Y|X = x\,]$ roven: $L(x) = w_H\left(S(X \oplus k^*)\right)$. [22]
- CPA je oproti NICV nižší, kdy útočník předpokládá špatný model, například $L(x) = w_H\,(x \oplus k^*)$, kdy $Y = w_H\left(S(x \oplus k^*)\right) + N$. [22]

Jestliže je X rozloženo rovnoměrně, není NICV samo o sobě rozlišovací prvek. Za předpokladu, že $Y = w_H\left(S(X \oplus k^*)\right) + N$, potom: $\mathrm{Var}\left[\mathbb{E}[Y|X|]\right] = \mathrm{Var}\left[w_H\left(S(X)\right)\right]$. Na druhou stranu, $Var\,[\,Y] = \mathrm{Var}\left[w_H\left(S(X)\right)\right] + Var[N]$. Vše dohromady tvoří vztah:

$$NICV = \frac{\mathrm{Var}\left[\mathbb{E}[Y|X|]\right]}{Var\,[\,Y]} = \frac{1}{1+\frac{1}{SNR}}\ , \qquad (3.2.4)$$

kde poměr signál-šum (SNR – Signal-to-noise ratio) je poměr mezi:

- signálem, což je rozptyl informativní části $Var\left[w_H\left(S(X \oplus k^*)\right)\right]$ a
- šumem, vyjádřeným jako $Var[N]$.

Je zřejmé, že výše uvedený vztah nezávisí na tajném klíči k^*, jelikož oba parametry X i Y jsou veřejné, a tudíž útočníkovi známé. Vzhledem k tomu, že NICV nezávisí na zvoleném klíči, nejedná se o nástroj pro vykonávání proudové analýzy, jinými slovy tato metoda sice ukáže všechny lineární závislosti veřejného parametru X s dostupnými průběhy Y (kde se nachází citlivé informace), ale už nedokáže určit útočníkem hledané hodnoty klíče (popř. užitých masek). Z tohoto důvodu je metoda NICV nejčastěji používána jako pomocný nástroj pro urychlení proudové analýzy (např. CPA).

3.3 Principal Component Analysis (PCA)

Pro potlačení šumu, či redukce dimenze dat, při zachování největšího rozptylu, je velmi často využívána analýza hlavních komponent (PCA). Jedná se o techniku předzpracování dat, kdy jsou pomocí lineární transformace původní proměnné nahrazeny novými tzv. hlavními komponentami. Tyto hlavní komponenty představují lineární kombinace původních proměnných a jsou sestupně seřazeny podle rozptylu, kdy první komponenta zachycuje největší rozptyl a poslední komponenta nejmenší rozptyl. Jinými slovy v první komponentě je obsažena největší část informace o rozptylu původních dat a nejmenší část informace obsahuje poslední komponenta. [23] [18]

PCA se především používá pro zmenšení dimenze dat, kdy je zmenšen počet znaků, aniž by došlo ke značné ztrátě informace. U DPA se analýza hlavních komponent používá z důvodu potlačení šumu, a tedy k usnadnění a urychlení samotné proudové analýzy. [18] [23]

3.3.1 Princip analýzy hlavních komponent (PCA)

Princip PCA se dá popsat několika těmito kroky [18] [23]:

Krok první: Získání dat

V první řadě musí útočník získat data, na které chce provést PCA. U proudové analýzy data představují měřené průběhy proudové spotřeby. [18] [23]

Krok druhý: Normalizace dat

Po získání dat je následně nutné je normalizovat, tzn. upravit je tak, aby se průměr každé dimenze rovnal 0. To se provádí tak, že od každého prvku dané dimenze se odečte její průměr. Jinými slovy v případě, že je k dispozici soubor dat o n rozměrech, je nezbytné od jednotlivého prvku daného rozměru odečíst průměr této dimenze: $\left(n_{ij} - \overline{n_i}\right)$, kde $\overline{n_i}$ značí průměr i-té dimenze a n_{ij} představuje j-tý prvek i-té dimenze. Takto vzniklá matice dat, jejíž průměr je roven 0, se nazývá matice normalizovaných dat. [18] [23]

Krok třetí: Výpočet kovarianční matice

Jestliže jsou data normalizovaná, je dále nutné vypočítat z nich kovarianční funkci, která vyjadřuje vztah mezi rozměry, a to sice podle výše uvedeného vztahu:

$$Cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \overline{X}) \cdot (Y_i - \overline{Y})}{(n-1)}, \tag{3.3.1}$$

kde X a Y značí jednotlivé dimenze a n představuje počet jejich prvků. Výsledek funkce kovariance může, podobně jako korelační koeficient, nabývat hodnot $-1 \leq Cov(X,Y) \leq 1$, kdy v případě, že se hodnota kovariance rovná právě -1, můžeme říct, že se jedná o zápornou vazbu mezi dimenzemi, kdy prvky v obou rozměrech zároveň klesají. Jestliže se kovarianční funkce rovná 1, jedná se o kladnou vazbu mezi dimenzemi, kdy prvky v obou rozměrech zároveň současně rostou. V případě, kdy je kovariance rovna 0, je mezi prvky dimenzí velice slabá, nebo dokonce žádná statisticky zjistitelná vazba.

Podle rovnic (3.1.3) a (3.1.5) lze usoudit, že prakticky rozptyl je kovariancí jedné dimenze. [18]

Kovarianční matice je za všech okolností čtvercová, tedy pokud je k dispozici soubor dat o n dimenzích, je nezbytné spočítat kovarianční matici mezi všemi těmito dimenzemi a tato matice bude mít rozměry $n \times n$. Kovarianční matice mezi 3 dimenzemi tedy bude mít rozměry 3×3 a bude symetrická podél hlavní diagonály. Jak již bylo řečeno kovariance jedné dimenze rozptyl, tudíž v hlavní diagonále kovarianční matice budou rozptyly daných dimenzí. Taková matice se třemi dimenzemi bude vypadat takto:[18][23]

$$C = \begin{pmatrix} Cov(n_1,n_1) & Cov(n_1,n_2) & Cov(n_1,n_3) \\ Cov(n_2,n_1) & Cov(n_2,n_2) & Cov(n_2,n_3) \\ Cov(n_3,n_1) & Cov(n_3,n_2) & Cov(n_3,n_3) \end{pmatrix}$$

Krok čtvrtý: Výpočet vlastního čísla a vlastního vektoru kovarianční matice

Vzhledem k tomu, že kovarianční matice je čtvercová, lze k této matici spočítat vlastní číslo a vektor. Vlastním vektorem matice je myšlen takový vektor, který při vynásobení matice tímto vektorem změní jen svoji velikost a nikoliv směr. Lze jej popsat vztahem:

$$A \cdot x = \lambda \cdot x, \tag{3.3.1}$$

kde A označuje čtvercovou matici, která má vlastní vektor x a λ označuje vlastní číslo vlastního vektoru x. Každý vlastní vektor má právě jedno vlastní číslo, což je koeficient, o který se při transformaci změní velikost tohoto vektoru. [18][23]

Avšak je důležité uvědomit si, že vlastní vektory může mít pouze čtvercová matice, a ne všechny čtvercové matice mají své vlastní vektory. Počet vlastních vektorů a čísel závisí na rozměrech čtvercové matice symetrické podle hlavní diagonály, tudíž v případě čtvercové matice o n rozměrech existuje n vypočitatelných vlastních vektorů a čísel. U analýzy hlavních komponent je navíc ještě nezbytné, aby vlastní vektory byly normalizovány, tedy jejich velikost byla rovna 1. [18][23]

Krok pátý: Sestavení vlastního vektoru dle vybraných komponent

Nyní je zapotřebí z matice sestupně seřazených vlastních vektorů vybrat ty s nejvyšší hodnotou vlastního čísla. Zpravidla jsou vybrány všechny vektory, jejichž vlastní číslo je větší nebo rovno 1. Vektor, jehož vlastní číslo má nejvyšší hodnotu, se nazývá hlavní komponenta daného souboru dat. Vybrané vlastní vektory jsou ve sloupcích následně opět sestupně seřazeny do nové matice, podle hodnot vlastního čísla. [18][23]

Krok šestý: Vytvoření nového datového souboru

Jako poslední je nezbytné extrahovat data do nového souboru, aby na ně bylo možné použít proudovou analýzu.

Jestliže je sestavena matice z vybraných vlastních vektorů (hlavních komponent), tzv. $Feature\ Vector$, nová data se vytvoří pomocí vztahu:

$$DataNew = (DataNormalized \times FeatureVector)^T, \qquad (3.3.2)$$

kde T označuje transponovanou matici. Transponovaná matice vznikne prohozením řádků a sloupců matice původní, $DataNormalized$ představují normalizovaná data a $DataNew$ je matice výhradně vybraných vektorů. Tyto vybrané vektory jsou vzájemně kolmé, tedy došlo k redukci dimenze. Takto vytvořená data již lze použít pro proudovou analýzu. [18][23]

Pokud by se ke vztahu (3.3.2) přičetly průměry (odečtené v kroku 1), tak za předpokladu že $Feature\ Vector$ obsahuje všechny hlavní komponenty, je možné dostat zpět originální soubor dat [18]:

$$DataOld = (DataNormalized \times FeatureVector)^T + Means, \qquad (3.3.3)$$

kde $Means$ označuje odečtené průměry a $DataOld$ původní soubor dat.

4 Výsledky studentské práce

V této kapitole je nejprve představena internetová soutěž DPA Contest a v této práci použité průběhy proudové spotřeby. Následně jsou znázorněny jednotlivé implementace metod NICV a PCA realizované ve vývojovém prostředí MATLAB.

4.1 DPA Contest

Soutěž DPA Contest (dále jen soutěž) pořádaná výzkumnou skupinou Digital Electronic Systems pod hlavičkou Communication & Electronics Department francouzské univerzity Télécom ParisTech je iniciativa vzniklá v reakci na potřebu nástroje, který by umožnil výzkumníkům objektivním způsobem mezi sebou porovnat sílu svých DPA algoritmů. [zdroj]

Aby mohla být objektivita takového porovnání jednoznačně prokazatelná, je třeba všem účastníkům poskytnout jednotná data a co nejpřesněji popsat metodu, kterou tato data byla získána. Každé kolo soutěže proto definuje:

- šifru, kterou je úkolem prolomit (včetně všech náležitých parametrů),
- matematický popis jejího algoritmu,
- implementaci tohoto algoritmu v programovacím jazyce,
- počítač (mikrokontrolér), jenž takový program provádí, tedy model a konfiguraci čipu, návrh plošného spoje, v němž je umístěn,
- osciloskop a další zařízení použitá pro získání dat exaktním postupem popisujícím i následný „postprocessing“ dat (např. zarovnání na časové ose)

a spoustu dalších parametrů, jejichž znalost by pro útok mohla (však nikoliv nezbytně) být byť třeba jen náznakem důležitá.

Sic se zdá být zřejmé, že už pouze výběr takové specifikace musel představovat úkol nesmírně obtížný, pro úspěšnosti DPA Contestu jako výzkumného projektu se ukázal být zcela zásadním. Průběhy elektrických veličin v obvodech s mikročipem měřené s časovým rozlišením na úrovni nanosekund nejsou data, jež by si mohl leckdo obstarat s běžným domácím vybavením, a tím pádem si mohou DPA side-chain leakage vyzkoušet i lidé např. mimo akademickou sféru.

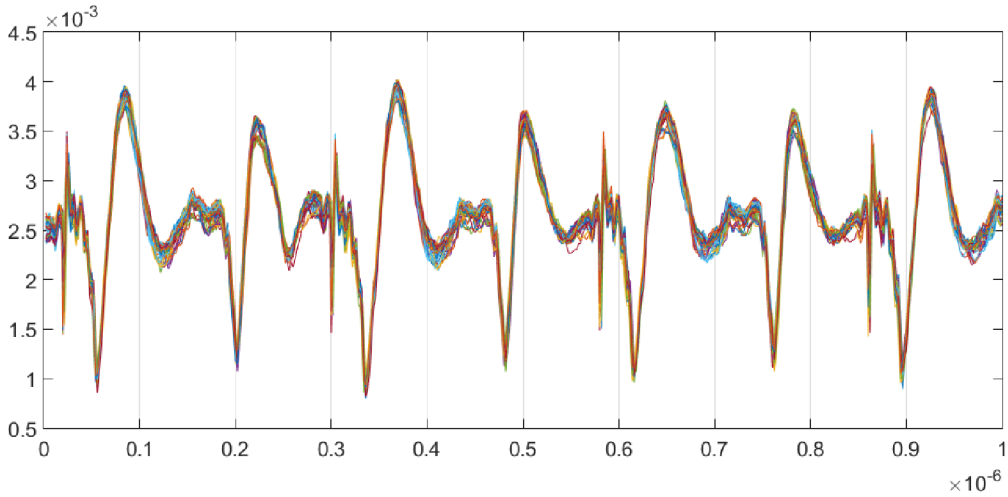
V současné době existuje již několik verzí této soutěže, v závislosti na implementaci šifrovacích algoritmů. Obě metody, jak NICV tak i PCA, jsou testovány na verzi DPA Contest v4, nicméně jsou navrženy tak, aby byly funkční a pro ostatní verze, zejména verzi v4.2, tedy za předpokladu, že samotné průběhy proudové spotřeby a soubor s indexem budou k dispozici ve stejném formátu, jako právě ve verzi v4.

4.2 Interpretace proudových průběhů a souboru s indexem

Proudové průběhy zvěřejněné se zadáním verze v4 představují průběhy proudové spotřeby mikrokontroleru ATMega 163 zachycené během 870 mikrosekund odpovídajícím délce celého procesu šifrování 8 bytů plaintextu použitým algoritmem AES-256 RSM.

Ke stažení jsou v ZIP archivech, čítajících 10 000 průběhů. Ač původní dataset obsahoval těchto ZIP archivů 10, spolu se zveřejněním verze v4.2 byl jejich počet snížen na 4, poněvadž nová data jsou daleko rozměrnější a server neposkytoval dostatek prostoru pro současné uložení všech dat. Protože verze v4 byla mnohokrát úspěšně prolomena s počtem průběhů řádově nižším, není důvod předpokládat, že by jejich absence činila někomu přítěž.

I přesto se však jedná o dataset nezanedbatelné velikosti, proto byly průběhy napřed zkomprimovány utilitou bzip2, s jejíž použitím byly z každého bajtu ušetřeny v průměru 4 bity, tedy zhruba polovina velikosti originálních dat, jejichž velikost po rozbalení z bzip2 archivů činí 16,4 GB.



Obr. 4.2: 100 průběhů proudové spotřeby

Na obr. 4.2 je znázorněno 100 průběhů proudové spotřeby internetové soutěže DPA Contest v4.

4.2.1 Binární formát záznamů proudových průběhů

Rozbalené soubory s průběhy jsou uloženy v binárním formátu WAVEDESC tak, jak byly pořízeny osciloskopem. Jeho struktura je vcelku jednoduchá, pro naše potřeby pak stačí vědět, že po 357 bajtech hlavičky začínají data kódovaná jako celá osmibitová čísla se znaménkem (int8) a pokračuji až po konec souboru.[CITACE]

Metadata včetně adresářové struktury nese 15MB dlouhý indexový soubor, na jehož řádcích jsou mezerou odděleny parametry jednotlivých měření:

- 32bajtový klíč,
- 16 bajtů otevřeného textu a k němu relevantních 16 bajtů šifrovaného textu,
- následované 4bitovým offsetem,
- číselným označením .zip archivu nesoucím odpovídající záznam průběhu a
- název takového souboru.

Binární hodnoty jsou vyjádřeny v šestnáctkovém kódu, pro snadné parsování mají všechny sloupce pevnou šířku.

Tato struktura dat byla použita jako výchozí pro další zpracování v systému MATLAB.

4.2.2 Načítání dat v systému MATLAB

Jak už sám název napovídá, v tomto vývojovém prostředí se nejpřirozeněji pracuje s maticemi, na které zde existuje nepřeberná řada funkcí a optimalizací. Protože pro optimální práci s maticemi je potřeba držet všechny její prvky v souvislém bloku paměti, efektivní práce s daty o rozměrech přesahujícím kapacitu operační paměti představuje netriviální problém.

Při analýze dat DPA contestu v4 není ovšem problém tento limit hravě překročit, a proto bylo v zájmu zachování obecnosti algoritmů navržených k naplnění cílů této bakalářské práce, opuštěno od standardních funkcí pro načítání vstupu, které nahradila vlastní implementace pracující na podobně nízké úrovni API.

Příkladem takové funkce budiž bzip2, která přečte bzip2 archiv uvedený v cestě a po jeho dekomprimaci vrátí obsah ve formě bajtů (int8). Sic MATLAB v základní výbavě bzip2 zpracovávat neumí, nahlédnutí do implementace jeho API pro práci s archivy formátu ZIP a GZIP, odhalující, že se opírá předně o třídy dostupné z Java Development Kitu, posloužila jako inspirace pro využití obdobného mechanizmu, s tím rozdílem však, že bzip2 knihovna pro Javu vyvíjená organizací Apache musela být do prostředí MATLABu explicitně načtena.

Po lepším seznámení se s aktuální dokumentací MATLABu se naskytla též zajímavá alternativa, a to využít verze matfilů (datových souborů s příponou .mat), které nevyžadují úplnou synchronizaci s RAM, nýbrž pouze těch bloků, u kterých se to zdá být nutné. Nespornou výhodu představuje fakt, že jde o vestavěnou schopnost MATLABu a poskytuje tedy jakousi formu záruky, co se týče stability a efektivity jejího využití.

To však, na druhou stranu, volá po výše zmíněném převodu dat do MATLAB-specifického formátu, a požadavek na ucelené řešení zůstává nenaplněn. Proto nakonec bakalářská práce uvádí obě varianty práce s daty.

4.3 Implementace NICV

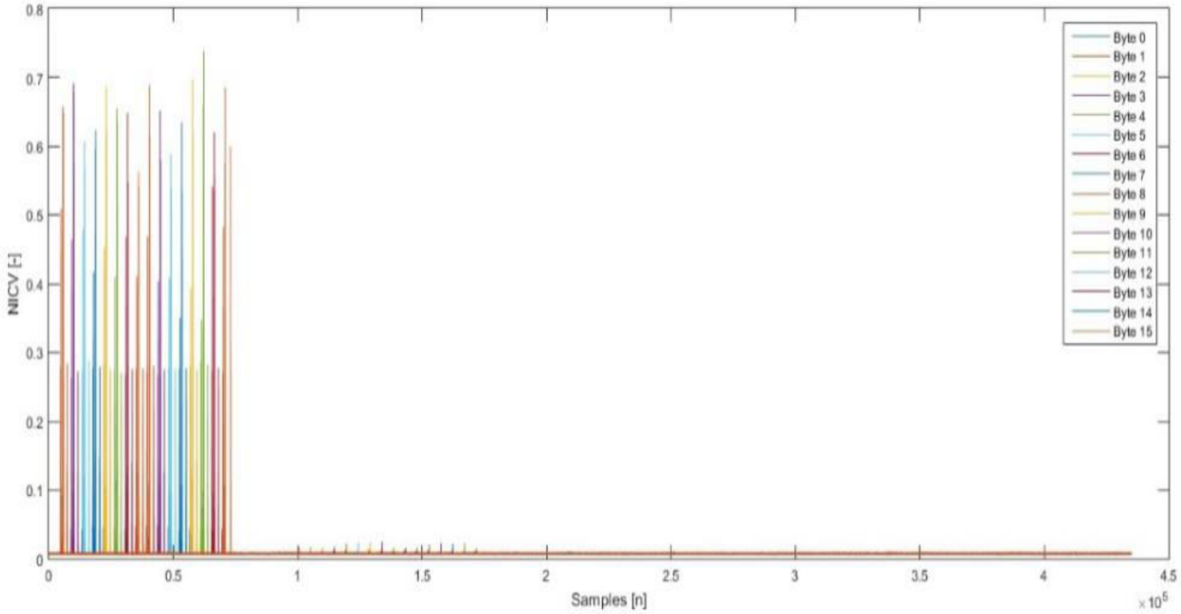
Výpočet NICV, tedy normalizovaného mezitřídního rozptylu je ze své matematické povahy triviální operací. Ze vztahu (4.2.1) vyplývá, že jde o poměr rozptylu mezi skupinami, tedy rozptylu vysvětleného (různou povahou skupin), ku rozptylu celkovému, tedy rozptylu mezi všemi proudovými průběhy.

Z hlediska implementace je tedy nutné pouze najít dělicí skupiny a rozdělit mezi ně naměřené stopy. Při aritmetických operacích je pak třeba brát zřetel na riziko ztráty platných cifer odčítáním floating-point datových typů s příliš odlišnou mantisou, což lze však ošetřit velmi snadno. Implementace v MATLABU proběhla podle následujícího empirického vztahu:

$$\widehat{NICV} = \frac{\frac{1}{2^n \cdot m}\sum_{x=0}^{2^n-1}\frac{\left(\sum_i s.t.X_i=x\, y_i\right)^2}{\sum_i s.t.X_i=x\, 1} - \left(\frac{1}{m}\sum_i y_i\right)^2}{\frac{1}{m}\sum_i y_i^2 - \left(\frac{1}{m}\sum_i y_i\right)^2} \qquad (4.2.1)$$

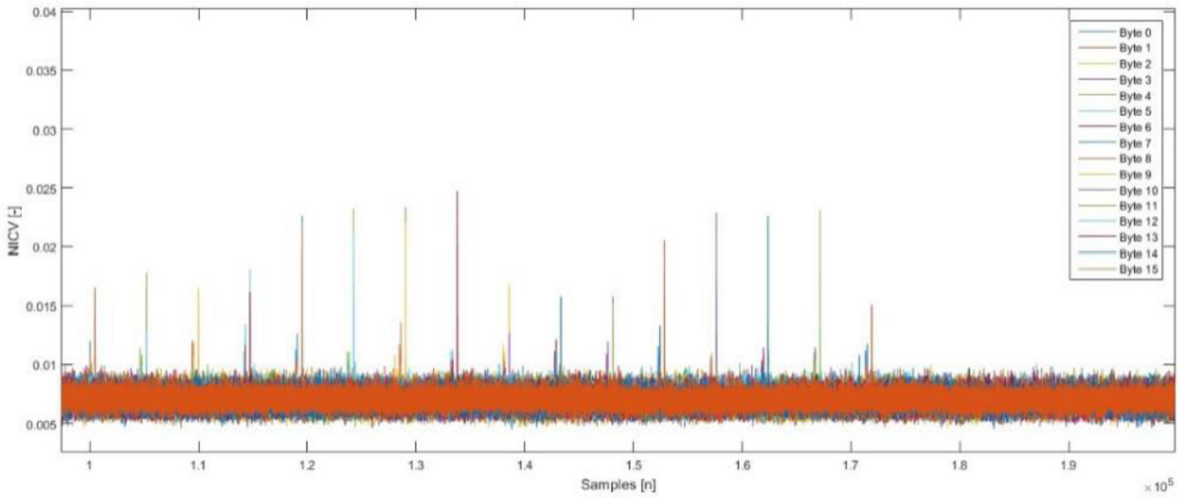
Výzvou pro co nejuniverzálnější implementací této funkce bylo nalezení takového algoritmu, jehož paměťová náročnost není lineární, nýbrž konstantní. Vhodnou úpravou vztahu pro NICV s využitím empirické formulace rozptylu, lze získat vztah, v kterém hodnota NICV závisí pouze na hodnotě malého počtu sumací odpovídajícímu počtu skupin navýšenému o 1.

Nejprve byla tato metoda počítána na všechny bajty otevřeného textu. Výpočet více jak jednoho bajtu je poměrně časově i výkonově náročné, ovšem záleží především na tom, zda je NICV počítáno pro celý dataset, nebo jen pro určitý počet průběhů.



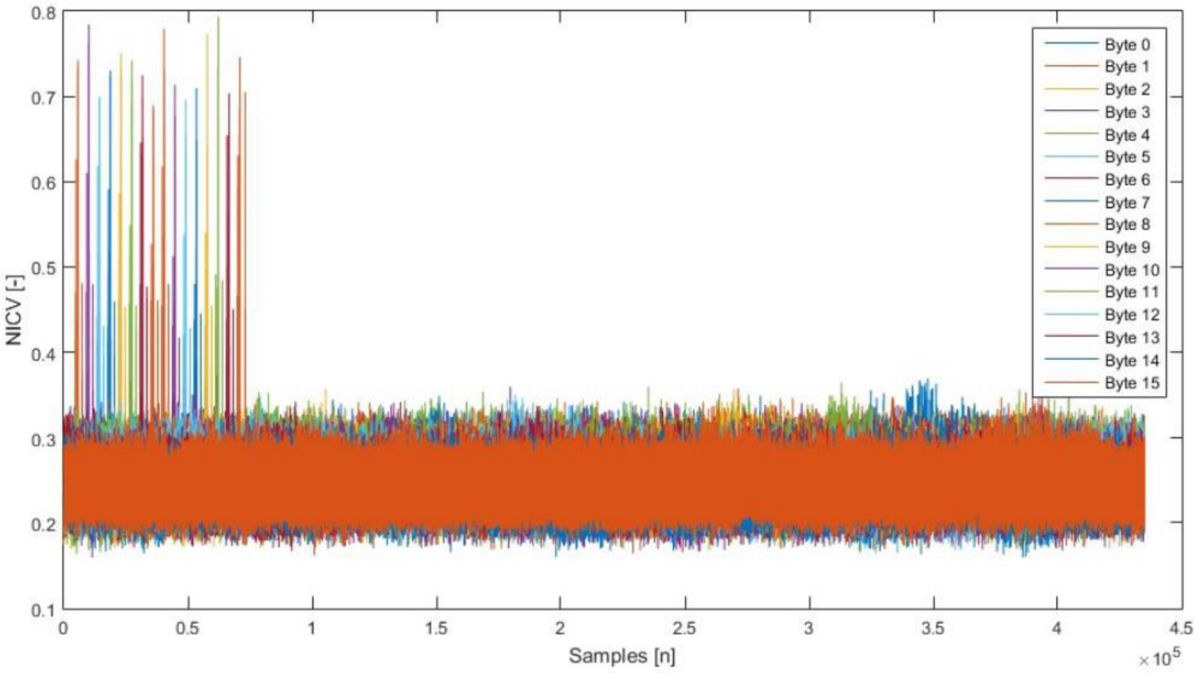
Obr. 4.2.1: NICV počítáno s ohledem na všechny bajty otevřeného textu pro celý dataset

Na obr. 4.2.1 lze vidět matici koeficientů NICV. Po pozornějším prozkoumání si lze všimnou malých závislostí mezi 100 000 – 200 000 průběhů. Podle [CITACE] tyto malé špičky odpovídají operaci SubBytes (substituce) algoritmu AES. Tyto informace již mohou být citlivé a zneužité útočníkem. Velké závislosti značí načítání otevřeného textu, tudíž pro útočníka jsou zcela bezcenné.



Obr. 4.2.2: Přiblížené malé špičky odpovídající operaci SubBytes

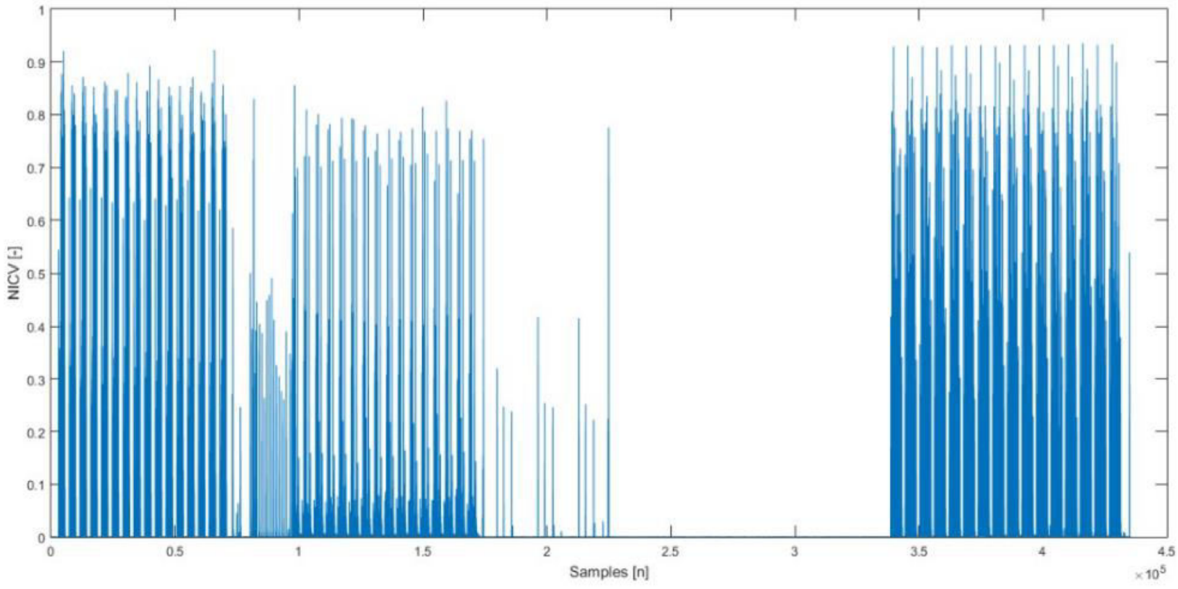
Počítání metody pro celý dataset a všechny bajty otevřeného textu je ovšem časově a paměťově velmi náročné, proto není od věci zjistit při jaké dolní mezi je útočník ještě schopen odchytit nějakou citlivou informaci a kdy je tato informace zcela skryta šumem.



Obr. 4.2.3: NICV počítáno pro 1000 průběhů proudové spotřeby

Obr. 4.2.3 ukazuje NICV počítané s ohledem na všechny bajty otevřeného textu, ovšem jen pro 1000 proudových průběhů. Sice je vidět závislost odpovídající načtení otevřeného textu, nicméně špičky odpovídající operaci SubBytes jsou zcela zašumělé.

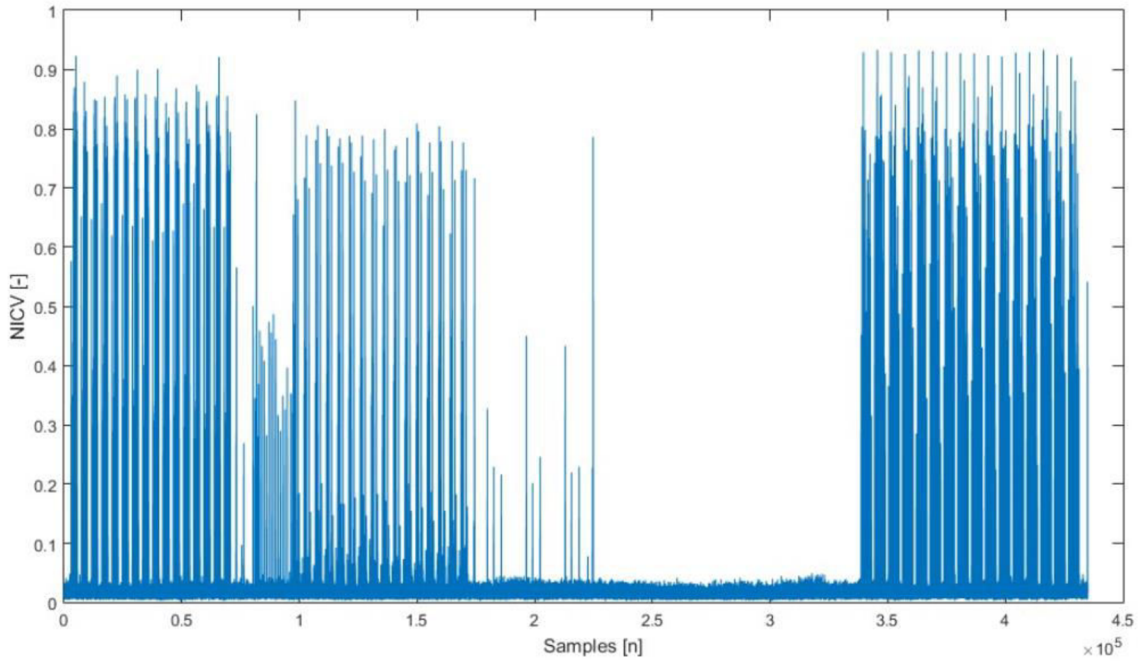
Jako další bylo NICV počítáno s ohledem na 4-bitový offset. Tento útok vzhledem v délce offsetu byl velmi rychlý, a to i při počítáním pro všech 40 000 průběhů.



Obr. 4.2.3: NICV počítáno s ohledem na 4-bitový offset na celý dataset

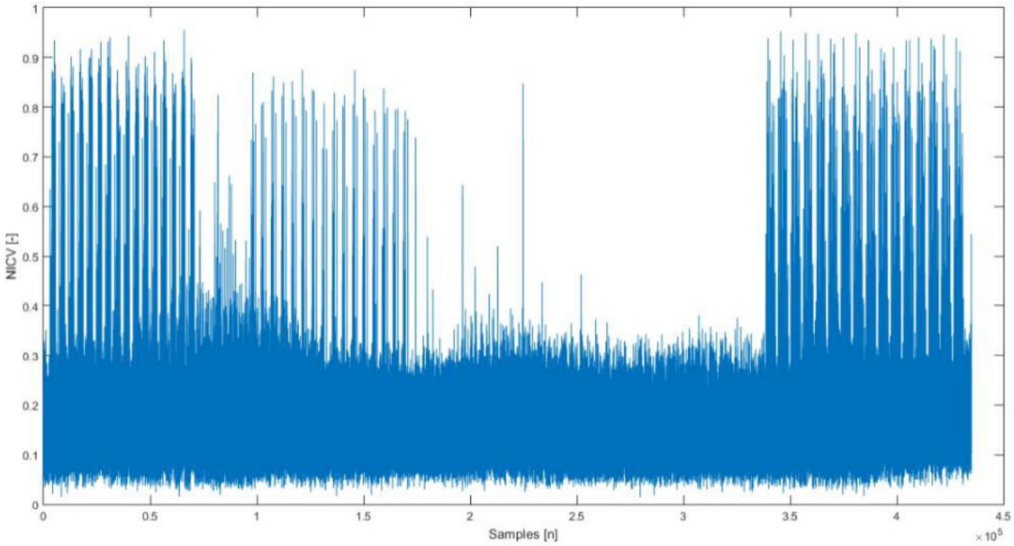
Obr. 4.2.3 ukazuje závislosti přímo vyzařující z offsetu. Dle [CITACE] se prvních 16 špiček týká operaci přidání masek u algoritmu AES, další série špiček pak odpovídá funkci RSM SubBytes a poslední špičky odpovídají korekci masky. Nyní už je schopen útočník zachytit velmi citlivou informaci a to je právě přidání masek (náhodná hodnota 0-255). NICV sice samo o sobě není schopno zjistit hodnoty masek, nicméně je to vynikající doplněk proudové analýzy (nejčastěji CPA).

Jen pro doplnění je spočítáno NICV s ohledem na offset i pro menší část průběhů.



Obr. 4.2.4: NICV počítáno s ohledem na 4-bitový offset pro 1000 průběhů proudové spotřeby

Na obr. 4.2.4 je NICV počítáno jen pro 1000 průběhů proudové spotřeby. Jak se lze přesvědčit hodnota šumu je i v tomto případě tak malá, že je zcela zřetelná lineární závislost a to i v případě, kdy je NICV počítáno pro 100 proudových průběhů, kde je sice šum znatelný, nicméně lineární závislost je stále zcela očividná.



Obr. 4.2.5: NICV počítáno s ohledem na 4-bitový offset pro 100 proudových průběhů

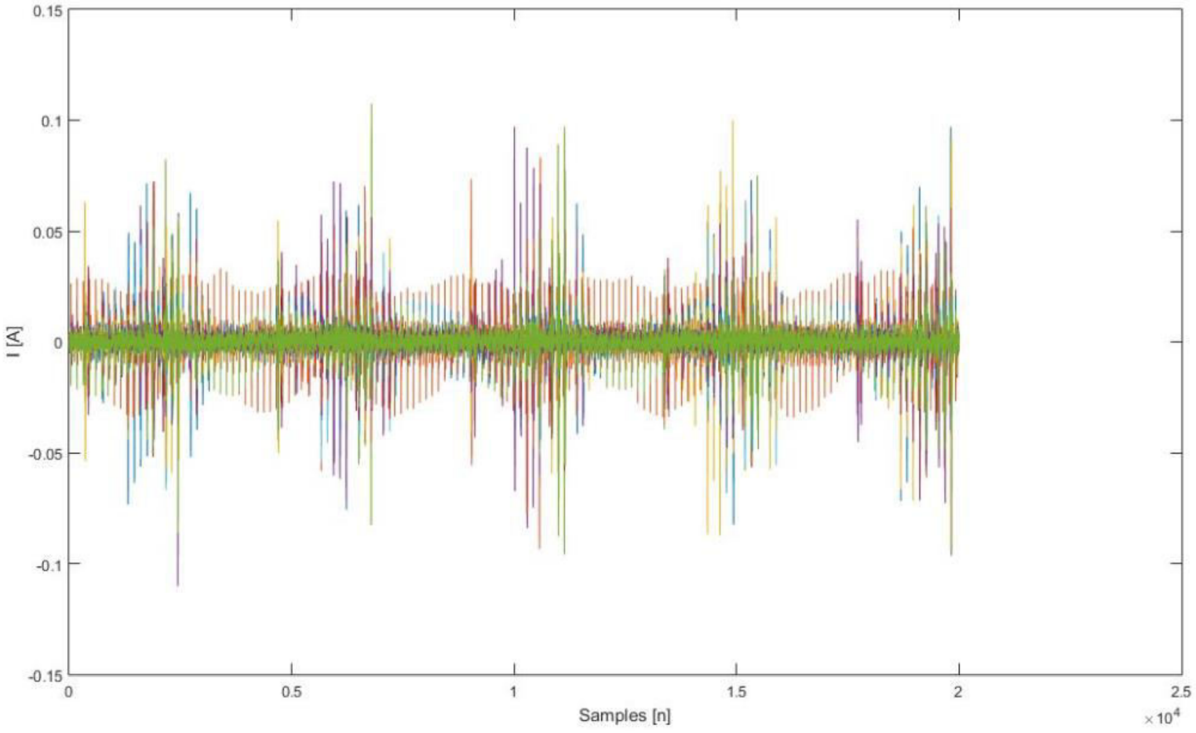
4.3 Implementace PCA

Metoda předzpracování dat PCA na rozdíl od metody NICV vůbec nepracuje se vstupními daty. Její úkol je pouze vzít proudové průběhy, zredukovat jejich dimenze a poté sestavit nové, na které je potom aplikována proudová analýza. V podstatě se jedná o komprimaci dat.

Co se týče matematické podstaty, tak jak již bylo znázorněno výše, CPA lze rozdělit do několika kroků. Načítání dat je provedeno stejně jako u metody NICV.

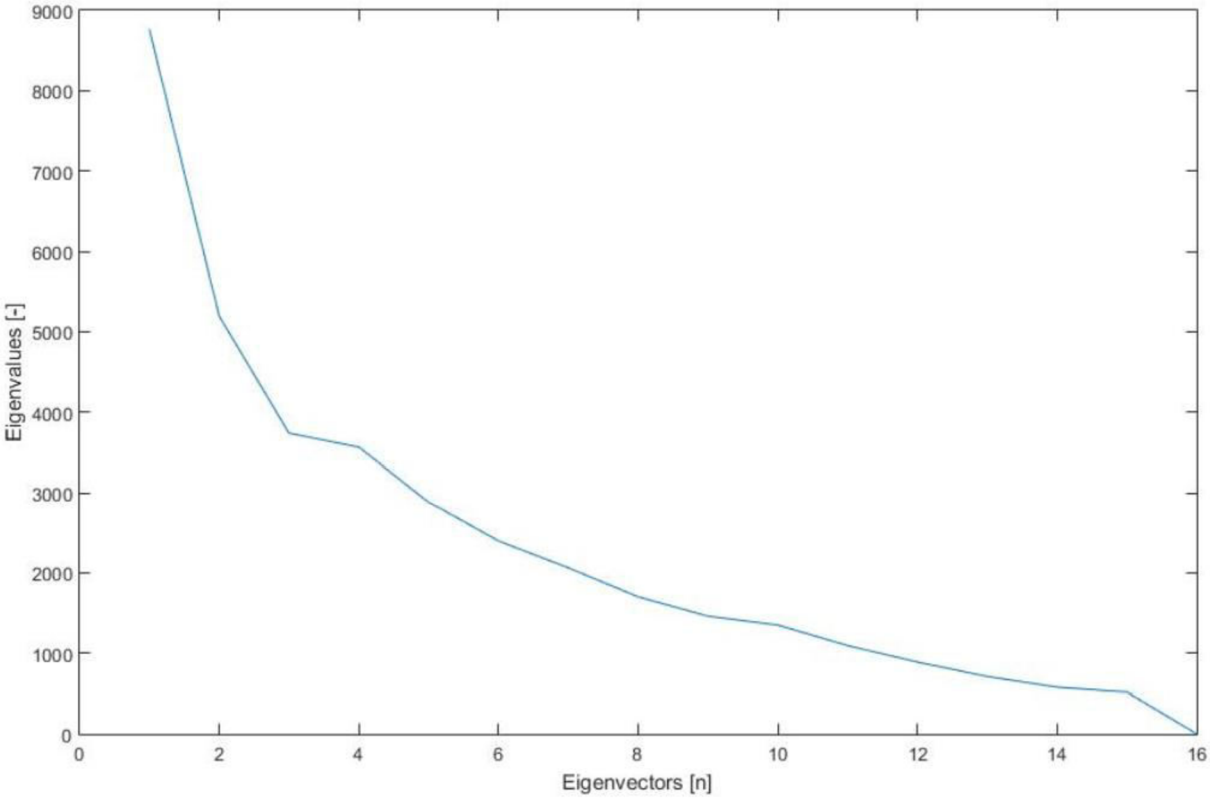
V první řadě je nutno podotknout, že metoda PCA je velmi časově a paměťově náročná a je tedy žádoucí neprovádět výpočet na celé proudové průběhy, ale spíše na jejich výseč.

Ze všeho nejdříve, po načtení dat, je nezbytné spočítat průměry skupin proudových průběhů a následně také celkový průměr. Následně je již proveden výpočet hlavních komponent a vlastních čísel.



Obr. 4.3.1: Matice 12ti hlavních komponent

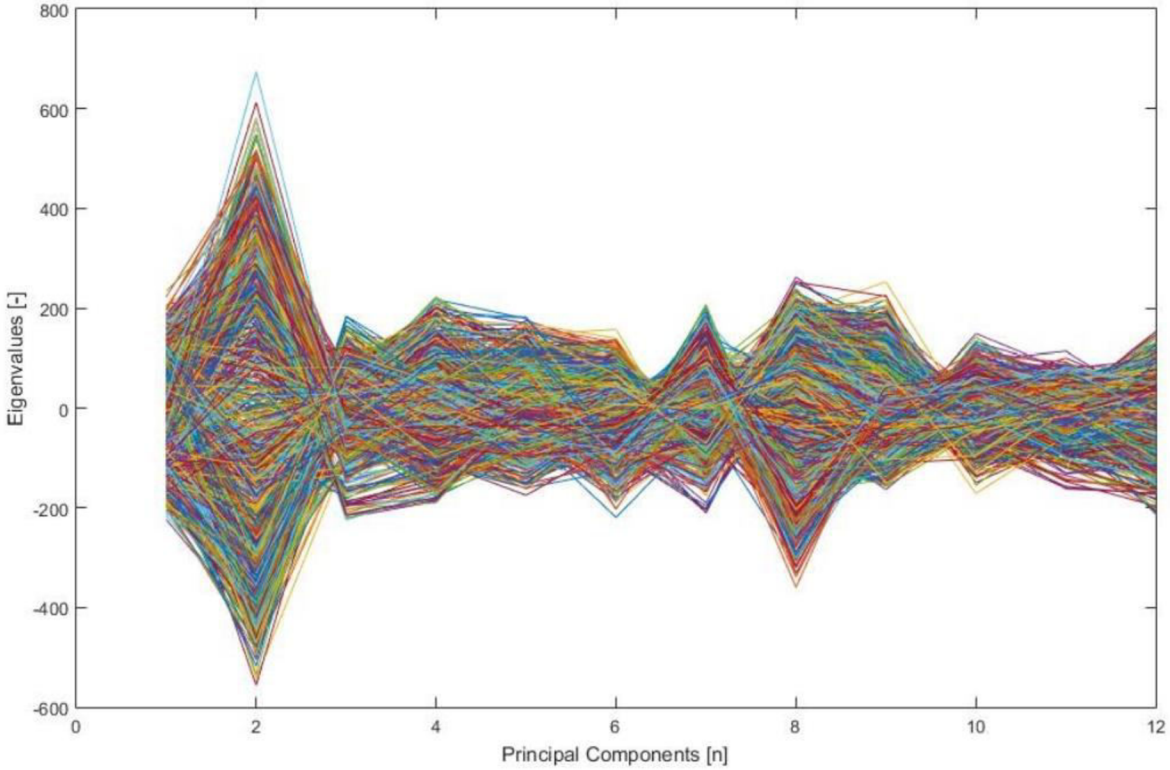
Komponenty jsou počítány dle percentilu, a to sice tak, že jsou vybrány pouze ty komponenty, jejichž percentil je větší jak 0,95. Na obr. 4.3.1 lze vidět 12 hlavních komponent, které byly vybrány. PCA bylo počítáno na 1000 průběhů proudové spotřeby a to z výseče 20 000 – 40 000 bodů.



Obr. 4.3.2: Matice vlastních vektorů a čísel

Díky této matici lze ověřit, že je výpočet hlavních komponent správný. Vlastnímu číslu s hodnotou 1000 (1) odpovídá právě 12 komponent.

Samotná data pak vzniknou vynásobením matice vlastních průměrů dat a výstupní matice.



Obr. 4.3.3: Výstupní matice

Vzniklá data jsou již připravena na proudovou analýzu, která by, za předpokladu správného výpočtu, měla odhalit správný klíč.

5 ZÁVĚR

Cílem této práce bylo zaměřit se na metody lokalizace zajímavých bodů u proudové analýzy. V teoretické části jsou představeny metody proudové analýzy, přičemž u diferenciální proudové analýzy je nastíněn postup při získání správného klíče. Následně jsou podrobně rozepsány principy samotných metod lokalizace zajímavých bodů, a to sice metody DPA založené na korelačním koeficientu (CPA), metody založené na analýze rozptylu (NICV) a metodou předzpracování dat (PCA).

V praktické části je nejdříve představena internetová soutěž DPA Contest a s ní i interpretace samotných průběhů proudové spotřeby, které byly použity při implementaci dvou vybraných metod, a to NICV a PCA. Tyto průběhy proudové spotřeby bylo nejdříve třeba šetrným způsobem importovat do prostředí MATLAB. Práce se zabývá dvojím způsobem načítání těchto průběhů, a to ve formátech .trc a .bzp2. Poté je implementována metoda NICV, díky které lze zjistit lineární závislosti ze zašifrovaných dat. Metoda PCA není sama o sobě metodou pro lokalizaci zajímavých bodů, pouze redukuje dimenze dat, které jsou následně podrobeny proudové analýze.

Obě metody byly testovány na proudové průběhy DPA Contest v4. Pro metodu NICV byla navíc napsána funkce, zajišťující načtení proudových průběhů i v kompresním tvaru bzip2, tudíž není nutné samotné průběhy proudové spotřeby gunzipovat.

Ze samotné implementace lze jednotlivých metod lze usoudit, že NICV ani PCA nejsou nástroji proudové analýzy, pouze ji jistým způsobem buď urychlují, nebo redukují vstupní data, což rovněž vede k větší rychlosti odhalení správného klíče.

LITERATURA

1. BHASIN, Shivam, Jean-Luc DANGER, Sylvain GUILLEY a Zakaria NAJM. *Side-Channel Leakage and Trace Compression using Normalized Inter-Class Variance* [online]. [cit. 2015-12-11]. Dostupné z: https://eprint.iacr.org/2013/717.pdf

2. BHASIN, Shivam, Nicolas BRUNEAU, Jean-Luc DANGER, Sylvain GUILLEY a Zakaria NAJM. *Analysis and Improvements of the DPA Contest v4 Implementation* [online]. [cit. 2015-12-12]. Dostupné z: http://www.dpacontest.org/v4/data/v4_2/article_implem_dpav42.pdf

3. BRIER. Eric, Christophe Clavier a Francis Olivier. *Correlation Power Analysis with a Leakage Model* [online]. [cit. 2015-12-12]. Dostupné z: https://www.iacr.org/archive/ches2004/31560016/31560016.pdf

4. BY STEFAN MANGARD, Elisabeth Oswald. *Power analysis attacks revealing the secrets of smart cards*. Online-Ausg. New York, N.Y: Springer, 2007. ISBN 978-038-7381-626.

5. DOGET, Julien, Emmanuel PROUFF, Matthieu RIVAIN a François-Xavier STANDAERT. Univariate side channel attacks and leakage modeling. *Journal of Cryptographic Engineering* [online]. 2011,**1**(2): 123-144 [cit. 2015-12-8]. DOI: 10.1007/s13389-011-0010-2. ISSN 2190-8508. Dostupné z: http://link.springer.com/10.1007/s13389-011-0010-2

6. *DPA Contest* [online]. [cit. 2015-12-13]. Dostupné z: http://www.dpacontest.org/home/

7. CHOUDARY, Omar a Markus G. KUHN. *Efficient Template Attacks* [online]. [cit. 2015-12-14]. Dostupné z: http://eprint.iacr.org/2013/770

8. KOCHER, P. C.; Jaffe, J.; Jun, B.: Differential Power Analysis. In *CRYPTO '99: Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, London, UK: Springer-Verlag, 1999, ISBN 3-540-66347-9, s. 388–397.

9. KOLÁČEK, Jan a Kateřina KONEČNÁ. *Jak pracovat s MATLABem* [online]. [cit. 2015-12-13]. Dostupné z: https://www.math.muni.cz/~kolacek/vyuka/vypsyst/navod.pdf

10. LOMNÉ, Victor, Emmanuel PROUFF a Thomas ROCHE. *Behind the scene of side channel attacks.* [online]. [cit. 2015-12-14]. Dostupné z: https://eprint.iacr.org/2013/794.pdf

11. MARTINÁSEK, Zdeněk. *Kryptoanalýza postranními kanály: Side channel cryptanalysis : zkrácená verze Ph.D Thesis*. [Brno: Vysoké učení technické], c2013, 31 s. ISBN 978-80-214-4786-8. Dostupné také z: https://dspace.vutbr.cz/handle/11012/32902

12. PROUFF, E., M. RIVAIN a R. BEVAN. Statistical Analysis of Second Order Differential Power Analysis. *IEEE Transactions on Computers* [online]. 2009, **58**(6): 799-811 [cit. 2015-12-14]. DOI: 10.1109/TC.2009.15. ISSN 0018-9340. Dostupné z: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4752810

13. THILLARD, Adrian, Emmanuel PROUFF a Thomas ROCHE. *Success through Confidence: Evaluating the Effectiveness of a Side-Channel Attack* [online]. : 21 [cit. 2015-12-14]. DOI: 10.1007/978-3-642-40349-1_2. Dostupné z: http://link.springer.com/10.1007/978-3-642-40349-1_2

14. ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB: začínáme se signály*. 1. vyd. Praha: BEN - technická literatura, 2006, 271 s. ISBN 80-730-0200-0.

15. ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB: tvorba uživatelských aplikací*. 1. vyd. Praha: BEN - technická literatura, 2004, 215 s. ISBN 80-7300-133-0.

16. ZAPLETAL, Ondřej. *Klasifikátory proudových otisků*. 2014, 59 l. Dostupné také z: https://dspace.vutbr.cz/handle/11012/32902

17. JAKUBÍKOVÁ, Radka. *REALIZACE ÚTOKU NA MASKOVANÝ ŠIFROVACÍ ALGORITMUS* [online]. 2015. [cit. 2016-05-31].

18. JEDLIČKA, František. *ANALÝZA HLAVNÍCH KOMPONENT V PROUDOVÉ ANALÝZ*E [online]. 2015. [cit. 2016-05-31].

19. Lejla Batina, Jip Hogenboom , Jasper G.J. van Woudenberg. Getting *More from PCA: First Results of Using Principal Component Analysis for Extensive Power Analysis* [online]. 2012. [cit. 2016-05-31].

20. MILDE, David. *ANALÝZA ROZPTYLU* [online]. 2011. [cit. 2016-05-31].

21. QUISQUATER, Jean-Jacques. *Principal and Independent Component Analysis for Crypto-systems with Hardware Unmasked Units* [online]. 2003. [cit. 2016-05-31].

22. Shivam Bhasin, Jean-Luc Danger, Sylvain Guilley, Zakaria Najm. *Side-Channel Leakage and Trace Compression using Normalized Inter-Class Variance* [online]. 2013. [cit. 2016-05-31].

23. WALCZYSKO, Martin. *ANALÝZA EEG SIGNÁLU POMOCÍ ANALÝZY HLAVNÍCH KOMPONENT (PCA)* [online]. 2008. [cit. 2016-05-31].

24. Základní metody diferenciální proudové analýzy. Brno: Fakulta elektrotechniky a komunikačních technologií VUT v Brně, 2013. ISSN 1213-1539.