Java karty a bezpečnost

4. výjezdní seminář z vyhledávání znalostí – Cikháj 2006

Jan Krhovják

Fakulta informatiky, Masarykova univerzita

Úvod

- Kryptografické čipové karty
- Technologie JavaCard & Open Platform
- Bezpečnost kryptografických čipových karet
- Útoky postranními kanály
- Poloautomatická klasifikace JavaCard operací
- Analýza naměřených vzorků
- Závěr a otevřené problémy

Kryptografické čipové karty

- Základní vlastnosti (kontaktní, bezkontaktní, atd.)
  - Procesor (8, 16, nebo 32 bitů)
  - HW akcelerace (např. pro DES, AES či modulární aritmetiku)
  - Paměti typu ROM (stovky kB), EEPROM (desítky kB) a RAM (jednotky kB)
  - Generátory náhodných a pseudonáhodných čísel
- Další požadované vlastnosti
  - Vysoká obtížnost padělání (vysoká úroveň miniaturizace)
  - Řízení přístupu k souborům (použití PINů)
- Typické aplikace
  - Autentizace uživatelů (typicky dvou-faktorová)
  - Uchovávání citlivých dat (např. kryptografických klíčů)

Kryptografické čipové karty

- Základní vlastnosti (kontaktní, bezkontaktní, atd.)
  - Procesor (8, 16, nebo 32 bitů)
  - HW akcelerace (např. pro DES, AES či modulární aritmetiku)
  - Paměti typu ROM (stovky kB), EEPROM (desítky kB) a RAM (jednotky kB)
  - Generátory náhodných a pseudonáhodných čísel
- Další požadované vlastnosti
  - Vysoká obtížnost padělání (vysoká úroveň miniaturizace)
  - Řízení přístupu k souborům (použití PINů)
- Typické aplikace
  - Autentizace uživatelů (typicky dvou-faktorová)
  - Uchovávání citlivých dat (např. kryptografických klíčů)

Kryptografické čipové karty

- Základní vlastnosti (kontaktní, bezkontaktní, atd.)
  - Procesor (8, 16, nebo 32 bitů)
  - HW akcelerace (např. pro DES, AES či modulární aritmetiku)
  - Paměti typu ROM (stovky kB), EEPROM (desítky kB) a RAM (jednotky kB)
  - Generátory náhodných a pseudonáhodných čísel
- Další požadované vlastnosti
  - Vysoká obtížnost padělání (vysoká úroveň miniaturizace)
  - Řízení přístupu k souborům (použití PINů)
- Typické aplikace
  - Autentizace uživatelů (typicky dvou-faktorová)
  - Uchovávání citlivých dat (např. kryptografických klíčů)

Technologie Java Card & Open Platform

- Java Card (specifikace z roku 1996) a verze 2.0 (z roku 1997)
  - V ROM implementována Java Virtual Machine
    - Umožňuje provádění podmnožiny byte-kódu
    - Řídí přístup ke všem zdrojům
  - Nijak neřeší problém nahrávání a instalace více apletů
  - V podstatě nepřenositelné aplikace/aplety
- Open Platform (specifikace z roku 1998) a verze 2.0 (z roku 1999)
  - Specifikuje zabezpečení nahrávání a instalace více apletů
  - Podpora více aplikací/apletů $\Rightarrow$ snadná přenositelnost
- Java Card 2.1 (specifikace z roku 1999)
  - Umožňuje navíc i podepisování kódu
  - Podpora verifikace byte-kódu, sandboxy, SW firewall, . . .

Technologie Java Card & Open Platform

- Java Card (specifikace z roku 1996) a verze 2.0 (z roku 1997)
  - V ROM implementována Java Virtual Machine
    - ★ Umožňuje provádění podmnožiny byte-kódu
    - ★ Řídí přístup ke všem zdrojům
  - Nijak neřeší problém nahrávání a instalace více apletů
  - V podstatě nepřenositelné aplikace/aplety
- Open Platform (specifikace z roku 1998) a verze 2.0 (z roku 1999)
  - Specifikuje zabezpečení nahrávání a instalace více apletů
  - Podpora více aplikací/apletů $\Rightarrow$ snadná přenositelnost
- Java Card 2.1 (specifikace z roku 1999)
  - Umožňuje navíc i podepisování kódu
  - Podpora verifikace byte-kódu, sandboxy, SW firewall, . . .

Technologie Java Card & Open Platform

- Java Card (specifikace z roku 1996) a verze 2.0 (z roku 1997)
  - V ROM implementována Java Virtual Machine
    - ★ Umožňuje provádění podmnožiny byte-kódu
    - ★ Řídí přístup ke všem zdrojům
  - Nijak neřeší problém nahrávání a instalace více apletů
  - V podstatě nepřenositelné aplikace/aplety
- Open Platform (specifikace z roku 1998) a verze 2.0 (z roku 1999)
  - Specifikuje zabezpečení nahrávání a instalace více apletů
  - Podpora více aplikací/apletů $\Rightarrow$ snadná přenositelnost
- Java Card 2.1 (specifikace z roku 1999)
  - Umožňuje navíc i podepisování kódu
  - Podpora verifikace byte-kódu, sandboxy, SW firewall, . . .

Útoky na kryptografické čipové karty

- Invazivní útoky
  - Přímý přístup ke komponentům čipu (sběrnice, paměť, . . .)
  - Mikrosondy, techniky čtení dat z paměti, . . .
  - Vyžadují mnoho času, znalostí, specializované vybavení
- Semi-invazivní útoky
  - Pouze odhalení (ale žádné poškození) čipu
  - UV či rentgenové záření, lokální zahřívání, el. mag. pole, . . .
  - Vyžadují pouze poměrně levné vybavení
- Neinvazivní útoky
  - Žádné fyzické poškození zařízení
  - Softwarové útoky
  - Odposlouchávání a monitorování
  - Útoky postranními kanály (TA, PA, FA, EMA)

Útoky na kryptografické čipové karty

- Invazivní útoky
  - Přímý přístup ke komponentům čipu (sběrnice, paměť, . . .)
  - Mikrosondy, techniky čtení dat z paměti, . . .
  - Vyžadují mnoho času, znalostí, specializované vybavení
- Semi-invazivní útoky
  - Pouze odhalení (ale žádné poškození) čipu
  - UV či rentgenové záření, lokální zahřívání, el. mag. pole, . . .
  - Vyžadují pouze poměrně levné vybavení
- Neinvazivní útoky
  - Žádné fyzické poškození zařízení
  - Softwarové útoky
  - Odposlouchávání a monitorování
  - Útoky postranními kanály (TA, PA, FA, EMA)

Útoky na kryptografické čipové karty

- Invazivní útoky
  - Přímý přístup ke komponentům čipu (sběrnice, paměť, . . .)
  - Mikrosondy, techniky čtení dat z paměti, . . .
  - Vyžadují mnoho času, znalostí, specializované vybavení
- Semi-invazivní útoky
  - Pouze odhalení (ale žádné poškození) čipu
  - UV či rentgenové záření, lokální zahřívání, el. mag. pole, . . .
  - Vyžadují pouze poměrně levné vybavení
- Neinvazivní útoky
  - Žádné fyzické poškození zařízení
  - Softwarové útoky
  - Odposlouchávání a monitorování
  - Útoky postranními kanály (TA, PA, FA, EMA)

Časová analýza

- Čas může u některých kryptooperací korelovat s hodnotami tajných klíčů
  - Často využívají modulární umocňování:

INPUT: $M, N, d = (d_{n-1}d_{n-2}\dots d_1 d_0)_2$
OUTPUT: $S = M^d \mod N$

1 $S \leftarrow 1$
2 for $j = n-1 \dots 0$ do
3 $\quad S \leftarrow S^2 \mod N$
4 $\quad$ if $d_j = 1$ then
5 $\quad\quad S \leftarrow S \cdot M \mod N$
6 return S

  - Původně Diffie-Hellman, RSA, DSS, ale také AES, IDEA, . . .
- Obrana
  - Použití operací zabírajících stejné množství času – neefektivní
  - Přidání šumu (zajistí náhodné délky operací) – DTA
  - Algoritmická protiopatření – nejlepší řešení

Časová analýza

- Čas může u některých kryptooperací korelovat s hodnotami tajných klíčů
  - Často využívají modulární umocňování:

INPUT: $M, N, d = (d_{n-1}d_{n-2}\dots d_1 d_0)_2$
OUTPUT: $S = M^d \mod N$

1 $S \leftarrow 1$
2 for $j = n-1 \dots 0$ do
3 $\quad S \leftarrow S^2 \mod N$
4 $\quad$ if $d_j = 1$ then
5 $\quad\quad S \leftarrow S \cdot M \mod N$
6 return S

  - Původně Diffie-Hellman, RSA, DSS, ale také AES, IDEA, . . .
- Obrana
  - Použití operací zabírajících stejné množství času – neefektivní
  - Přidání šumu (zajistí náhodné délky operací) – DTA
  - Algoritmická protiopatření – nejlepší řešení

Vkládání chyb do výpočtu

- Vkládání HW chyb může ovlivnit bezpečnost algoritmu
  - Jediný chybný podpis pomocí CRT RSA $\Rightarrow$ odhalení klíče
  - Chyby typu: změna hodinového taktu, změna dodávky energie
- Častá obrana – kontrola výstupů algoritmu
  - Velmi náročná (degradace výkonu)
  - Nemusí být dostačující
- Návrh algoritmů odolných proti chybám
  - Použití detekčních a opravných kódů
  - Redundantní aritmetika v konečných polích

Vkládání chyb do výpočtu

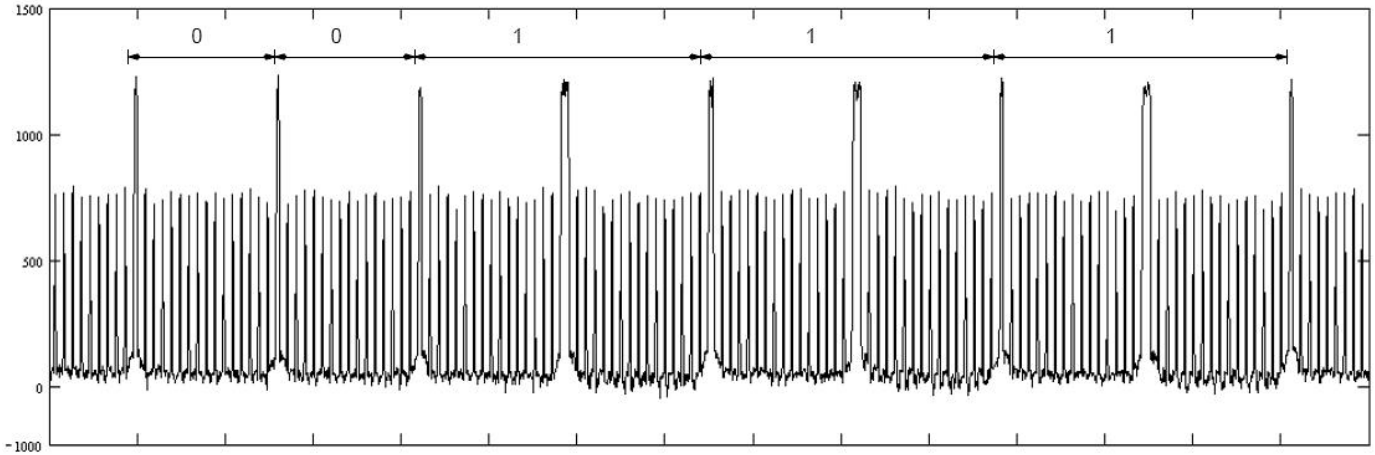
- Vkládání HW chyb může ovlivnit bezpečnost algoritmu
  - Jediný chybný podpis pomocí CRT RSA $\Rightarrow$ odhalení klíče
  - Chyby typu: změna hodinového taktu, změna dodávky energie
- Častá obrana – kontrola výstupů algoritmu
  - Velmi náročná (degradace výkonu)
  - Nemusí být dostačující
- Návrh algoritmů odolných proti chybám
  - Použití detekčních a opravných kódů
  - Redundantní aritmetika v konečných polích

Vkládání chyb do výpočtu

- Vkládání HW chyb může ovlivnit bezpečnost algoritmu
  - Jediný chybný podpis pomocí CRT RSA $\Rightarrow$ odhalení klíče
  - Chyby typu: změna hodinového taktu, změna dodávky energie
- Častá obrana – kontrola výstupů algoritmu
  - Velmi náročná (degradace výkonu)
  - Nemusí být dostačující
- Návrh algoritmů odolných proti chybám
  - Použití detekčních a opravných kódů
  - Redundantní aritmetika v konečných polích

Odběrová analýza

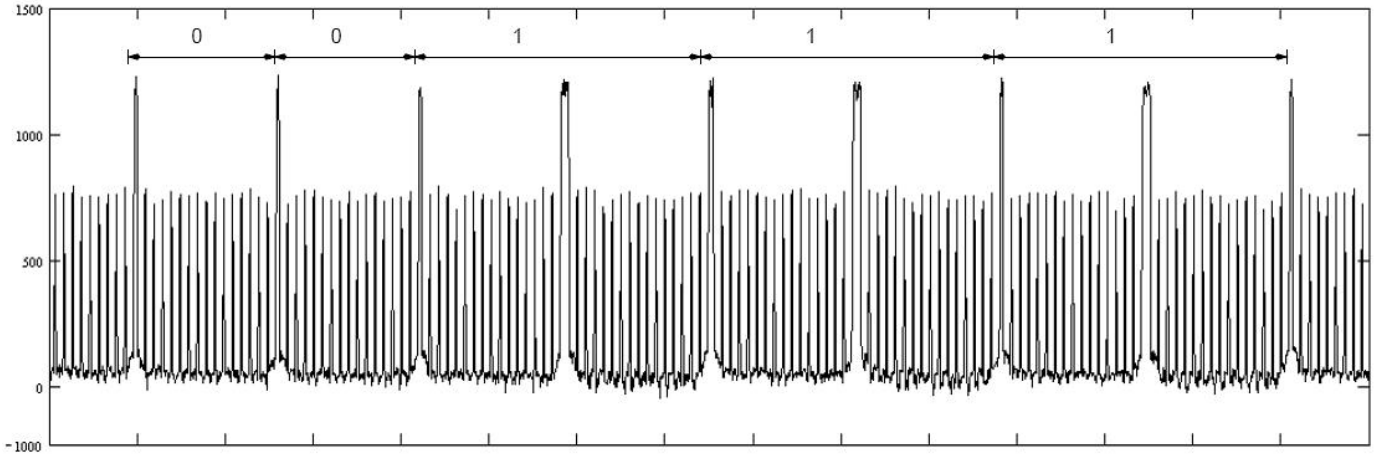
- Využití informace o množství spotřebované energie
- SPA – přímé vyhodnocení množství spotřebované energie
  - SPA vzorek provádění podpisu pomocí RSA $\Rightarrow$ 5 bitů klíče



- DPA – využití statistických metod (účinnější a nebezpečnější)
  - Odhalí i nepatrné výkyvy ve spotřebě energie
  - Eliminuje chyby při měření a šum
- Obrana – SW (náhodné maskování) či HW (nepřímé napájení)

Odběrová analýza

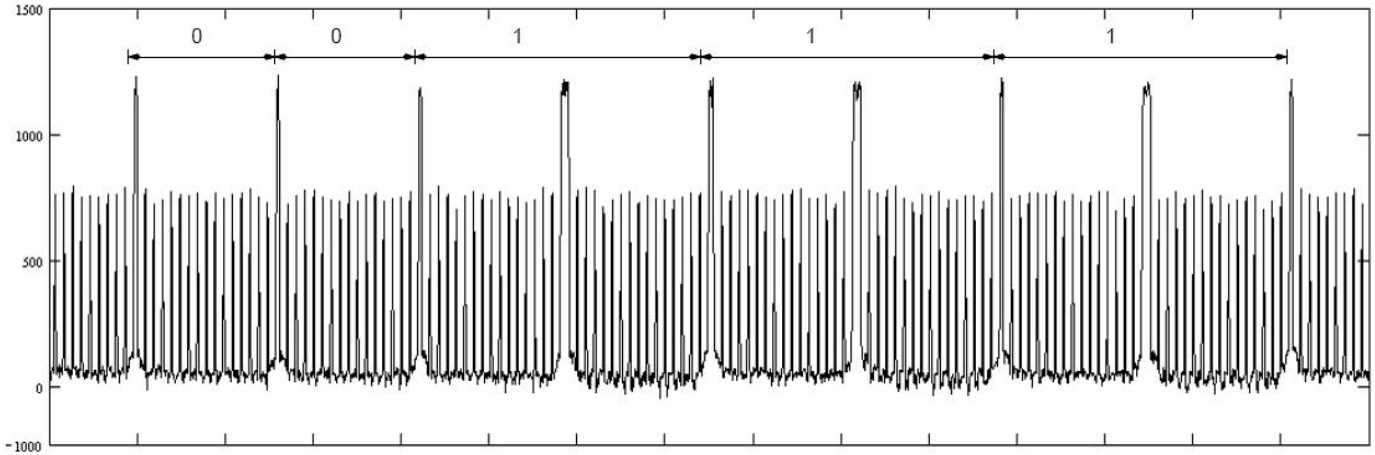
- Využití informace o množství spotřebované energie
- SPA – přímé vyhodnocení množství spotřebované energie
  - SPA vzorek provádění podpisu pomocí RSA $\Rightarrow$ 5 bitů klíče



- DPA – využití statistických metod (účinnější a nebezpečnější)
  - Odhalí i nepatrné výkyvy ve spotřebě energie
  - Eliminuje chyby při měření a šum
- Obrana – SW (náhodné maskování) či HW (nepřímé napájení)

Odběrová analýza

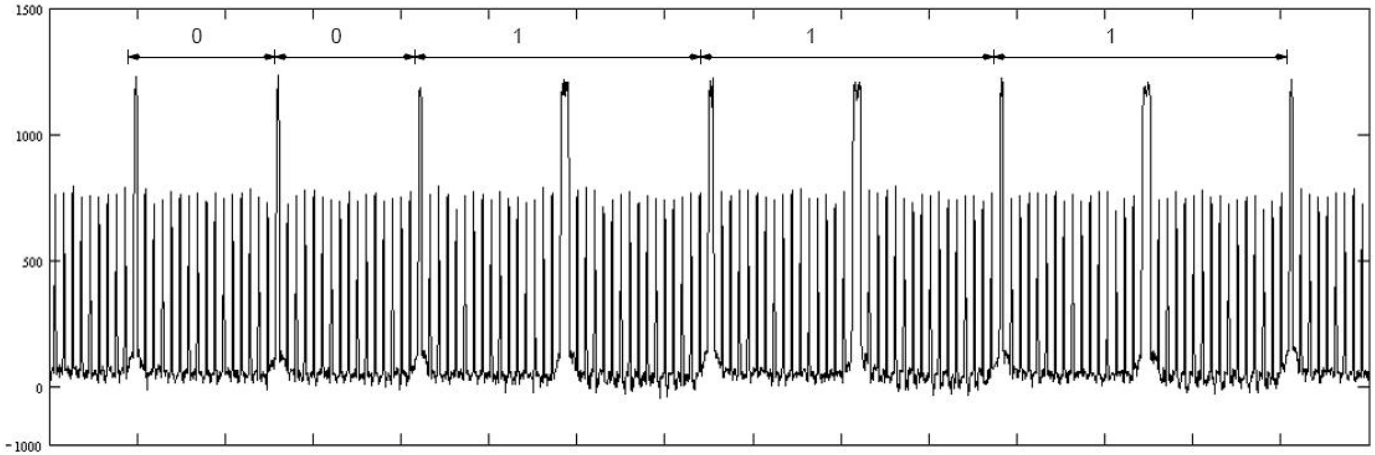
- Využití informace o množství spotřebované energie
- SPA – přímé vyhodnocení množství spotřebované energie
  - SPA vzorek provádění podpisu pomocí RSA $\Rightarrow$ 5 bitů klíče



- DPA – využití statistických metod (účinnější a nebezpečnější)
  - Odhalí i nepatrné výkyvy ve spotřebě energie
  - Eliminuje chyby při měření a šum
- Obrana – SW (náhodné maskování) či HW (nepřímé napájení)

Odběrová analýza

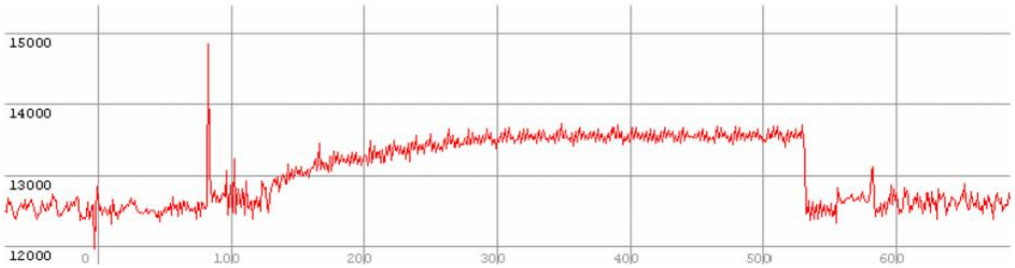
- Využití informace o množství spotřebované energie
- SPA – přímé vyhodnocení množství spotřebované energie
  - SPA vzorek provádění podpisu pomocí RSA $\Rightarrow$ 5 bitů klíče



- DPA – využití statistických metod (účinnější a nebezpečnější)
  - Odhalí i nepatrné výkyvy ve spotřebě energie
  - Eliminuje chyby při měření a šum
- Obrana – SW (náhodné maskování) či HW (nepřímé napájení)

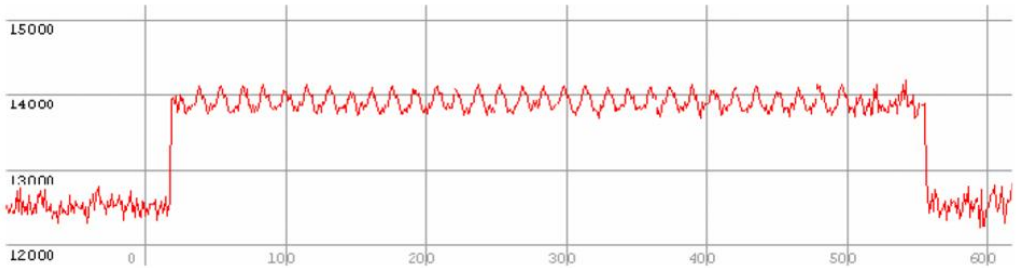
Poloautomatická klasifikace JavaCard operací

- Vytvoření databáze operací dle profilu příkonu
  - Posloupnost charakteristických vzorů
  - Logická struktura operace



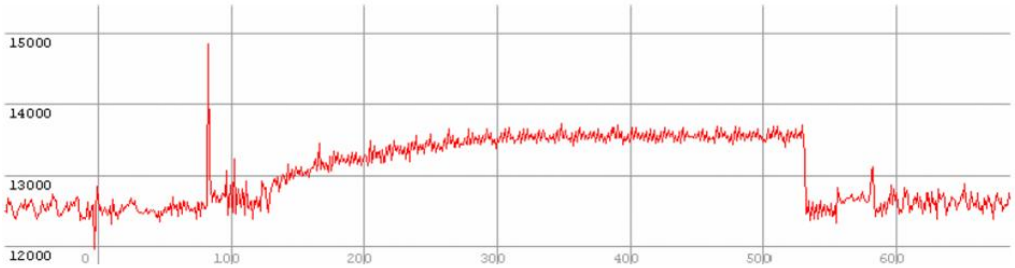
  - Algoritmus ověřování PINu (sniž→ověř→zvyš)

- Časový nedeterminismus
  - MD5, SHA1 – deterministické
  - Šifrovací operace, generování náhodných dat – nedeterministické



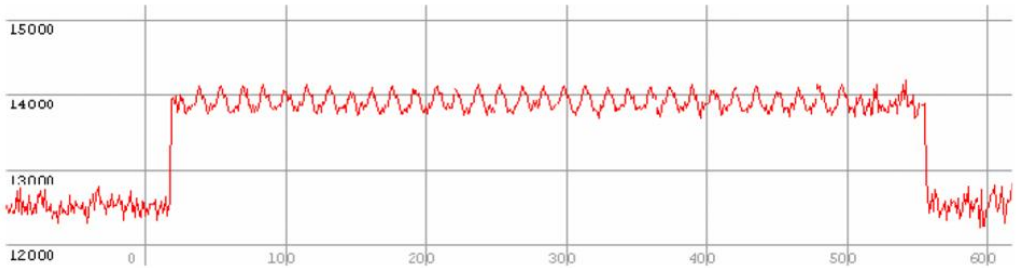
Poloautomatická klasifikace JavaCard operací

- Vytvoření databáze operací dle profilu příkonu
  - Posloupnost charakteristických vzorů
  - Logická struktura operace



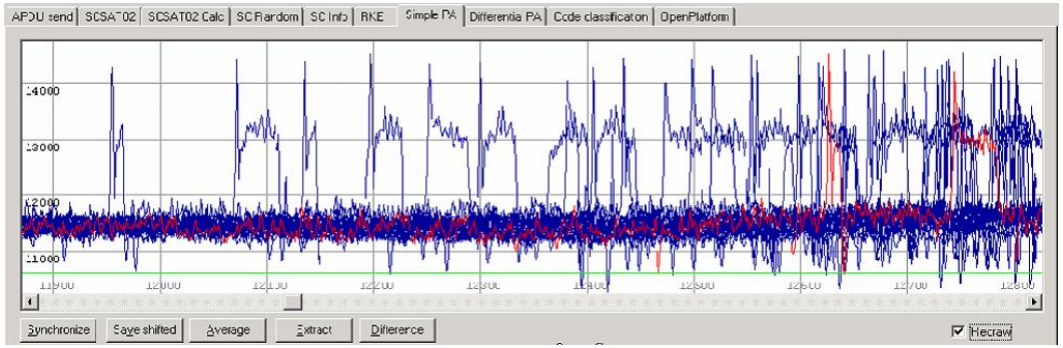
  - Algoritmus ověřování PINu (sniž→ověř→zvyš)

- Časový nedeterminismus
  - MD5, SHA1 – deterministické
  - Šifrovací operace, generování náhodných dat – nedeterministické

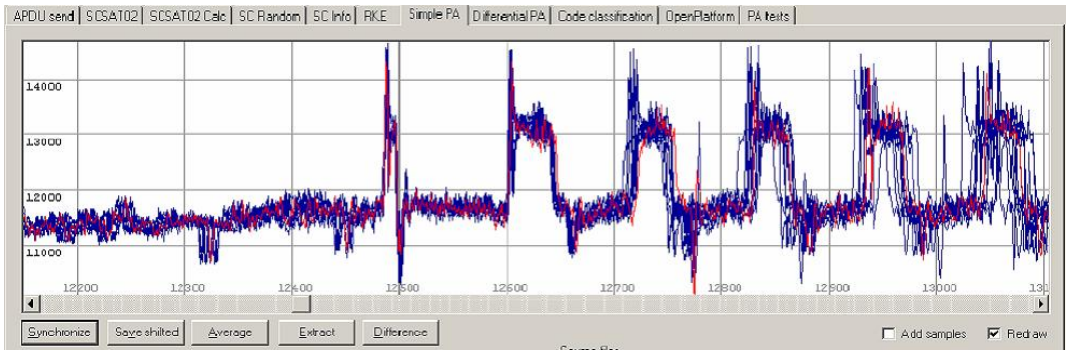


Analýza naměřených vzorků

- Nalezení pozice zkoumané operace, srovnání pro různá data
- Časový nedeterminismus některých kryptografických operací

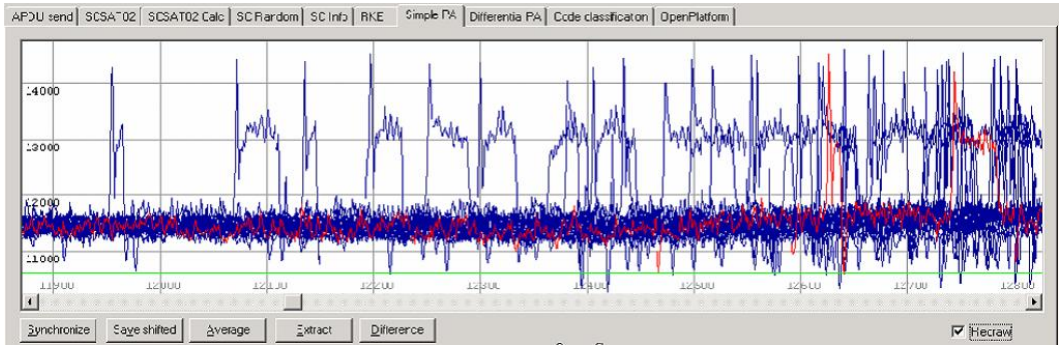


- Jak pak ale porovnávat měření (např. na 1000 vzorcích)?
  - Zkoumaná operace ani nezačíná vždy na stejném místě
  - Nutnost synchronizace – prozatím pouze poloautomatická

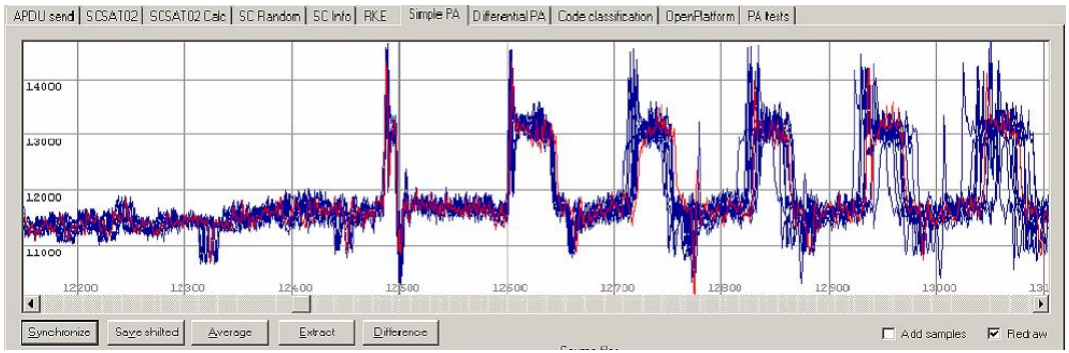


Analýza naměřených vzorků

- Nalezení pozice zkoumané operace, srovnání pro různá data
- Časový nedeterminismus některých kryptografických operací

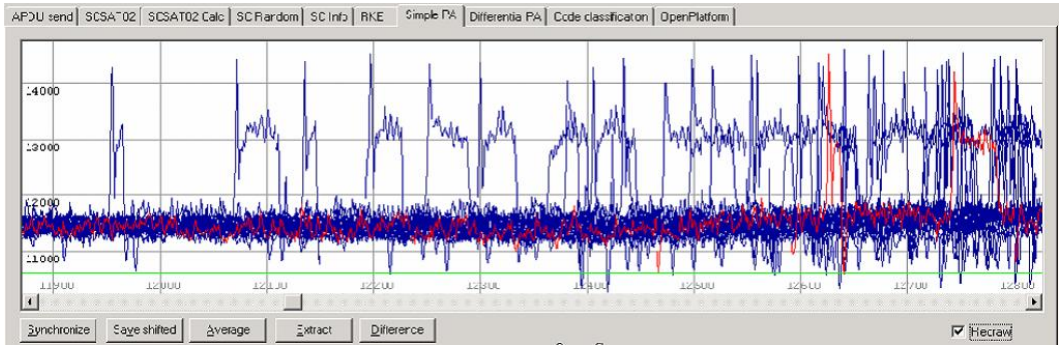


- Jak pak ale porovnávat měření (např. na 1000 vzorcích)?
  - Zkoumaná operace ani nezačíná vždy na stejném místě
  - Nutnost synchronizace – prozatím pouze poloautomatická

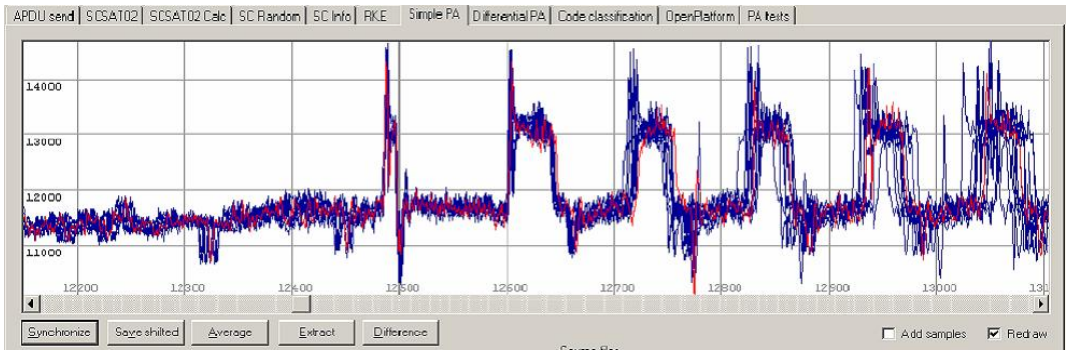


Analýza naměřených vzorků

- Nalezení pozice zkoumané operace, srovnání pro různá data
- Časový nedeterminismus některých kryptografických operací



- Jak pak ale porovnávat měření (např. na 1000 vzorcích)?
  - Zkoumaná operace ani nezačíná vždy na stejném místě
  - Nutnost synchronizace – prozatím pouze poloautomatická



Závěr a otevřené problémy

. . . DISKUZE . . .