riscure

Inspector 4.11

SCA & FI software update

December 2016

Contents

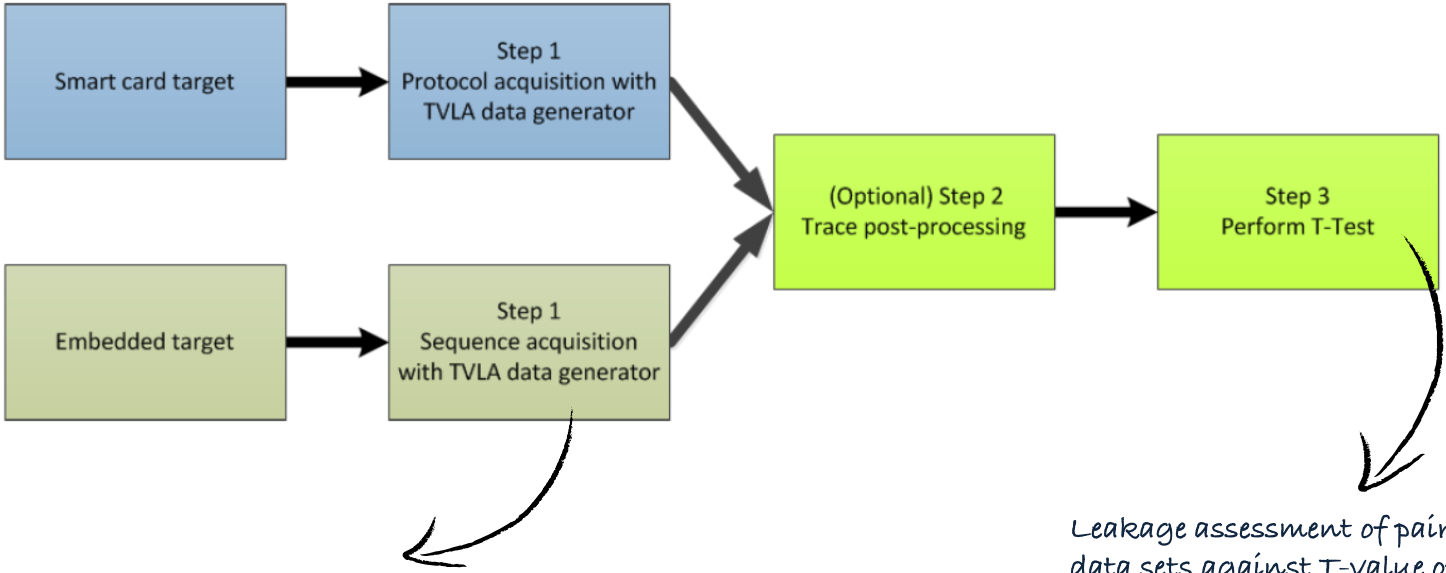
What’s new in 4.11?

Test Vector Leakage Assessment

Introducing TVLA in Inspector with

- first order analysis on DES & AES
- semi-constant (a.k.a. semi-fixed) round intermediates
- non-specific T-test

Workflow:



Smart card target

Step 1
Protocol acquisition with
TVLA data generator

Embedded target

Step 1
Sequence acquisition
with TVLA data generator

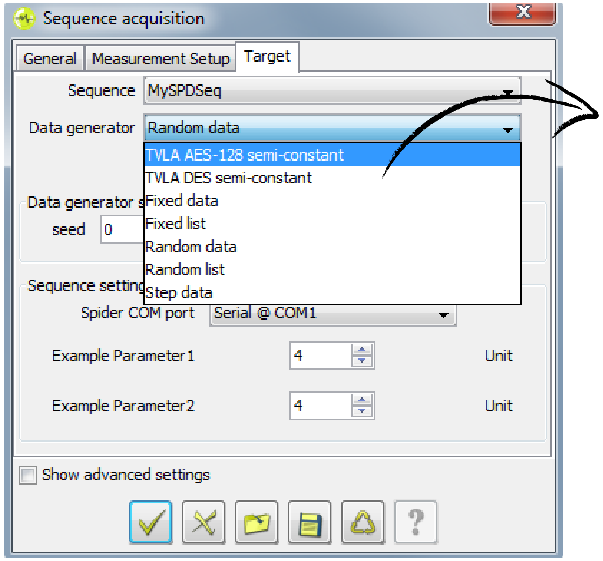
(Optional) Step 2
Trace post-processing

Step 3
Perform T-Test

Creating multiple test vectors at
once for efficiency

Leakage assessment of pairs of
data sets against T-value of 4.5

TVLA: acquisition with test vector data sets



Sequence: MySPDSeq

Data generator: Random data

- TVLA AES-128 semi-constant
- TVLA DES semi-constant
- Fixed data
- Fixed list
- Random data
- Random list
- Step data

Data generator s

seed: 0

Sequence settin

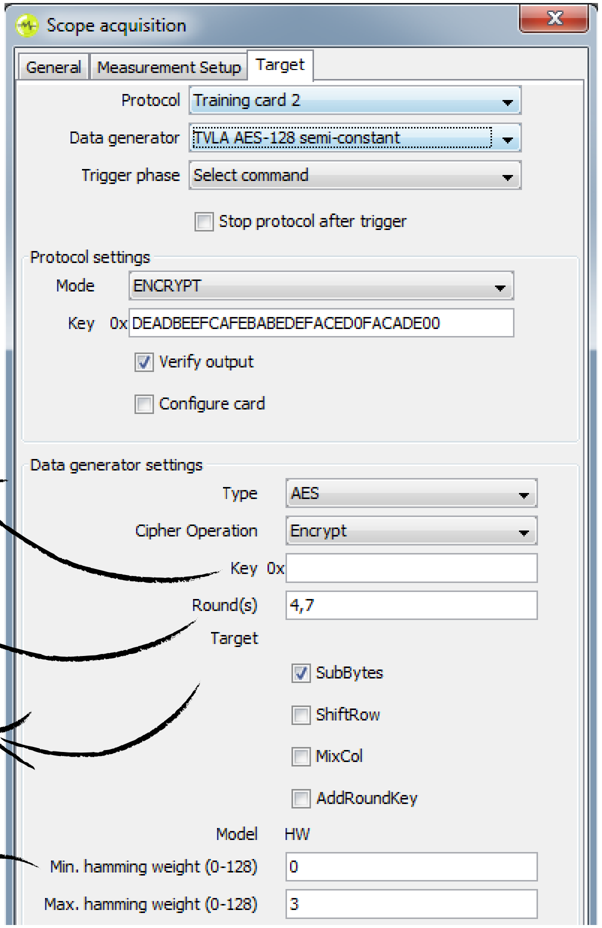
Spider COM port: Serial @ COM1

Example Parameter1: 4 Unit

Example Parameter2: 4 Unit

□ Show advanced settings

Generate semi-constant data set for DES or AES



Protocol: Training card 2

Data generator: TVLA AES-128 semi-constant

Trigger phase: Select command

□ Stop protocol after trigger

Protocol settings

Mode: ENCRYPT

Key 0x: DEADBEEFCAFEBABEDEFACED0FACADE00

☑ Verify output

□ Configure card

Data generator settings

Type: AES

Cipher Operation: Encrypt

Key 0x:

Round(s): 4,7

Target

☑ SubBytes

□ ShiftRow

□ MixCol

□ AddRoundKey

Model: HW

Min. hamming weight (0-128): 0

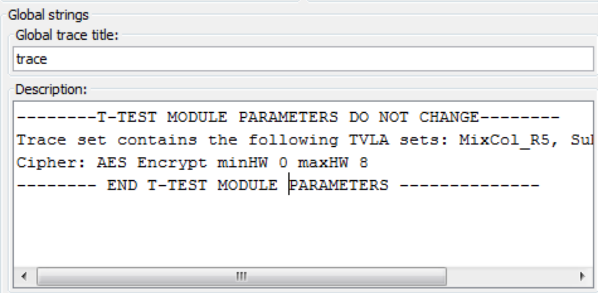
Max. hamming weight (0-128): 3

Enter fixed key

Round selection

Creates data sets for each target option (per selected AES round)

When saving a trace set: info on TVLA data sets


Global strings

Global trace title:

trace

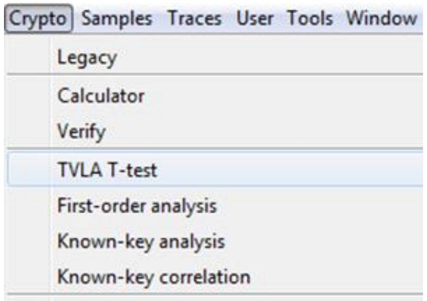
Description:

```
--------T-TEST MODULE PARAMETERS DO NOT CHANGE--------
Trace set contains the following TVLA sets: MixCol_R5, Sul
Cipher: AES Encrypt minHW 0 maxHW 8
-------- END T-TEST MODULE PARAMETERS --------------
```

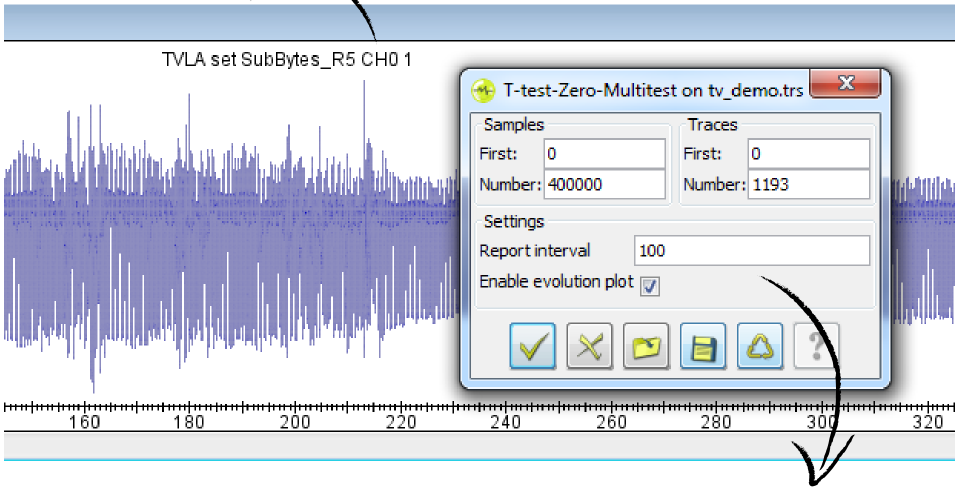
Semi-constant: Limited range ensures optimal variation of constant inputs for a meaningful T-test

TVLA: Welch’s T-test with real-time evolution plot

Run the non-specific T-test on the trace set with test vectors created with the data generator:



Target: Round 5 and Subbytes (AES)


TVLA set SubBytes_R5 CH0 1
T-test-Zero-Multitest on tv_demo.trs
160
180
200
220
240
260
280
300
320

Samples First: 0

Samples Number: 400000

Traces First: 0

Traces Number: 1193

Settings

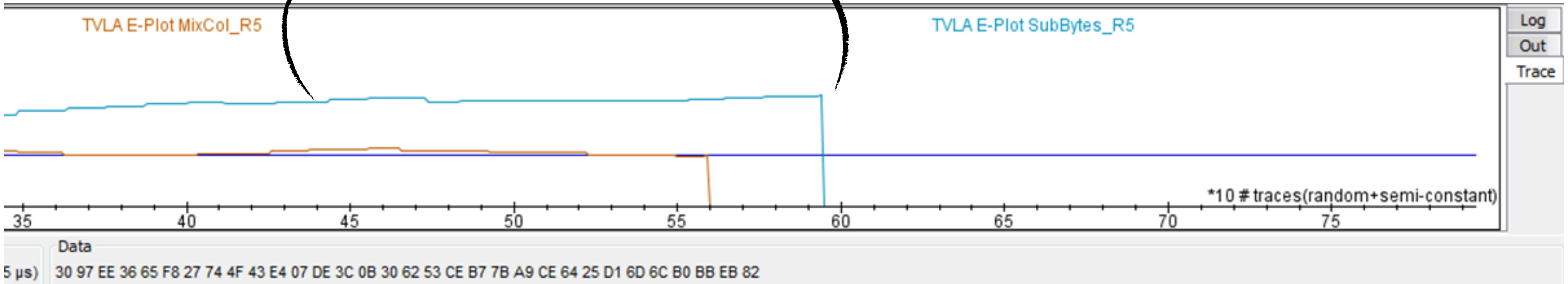
Report interval 100

Enable evolution plot ☑

Select samples and report interval for T-test

Monitor evolution plot real time

Abort without loosing the results



Data

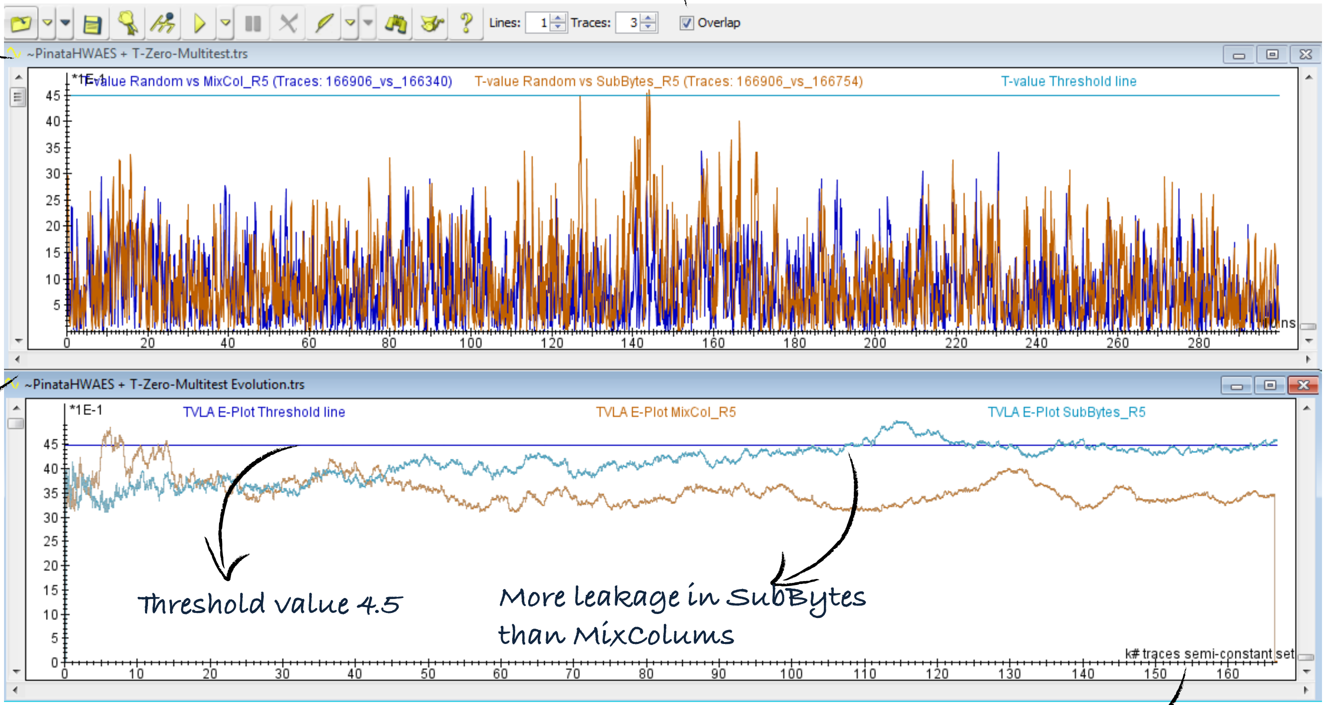
5 µs) 30 97 EE 36 65 F8 27 74 4F 43 E4 07 DE 3C 0B 30 62 53 CE B7 7B A9 CE 64 25 D1 6D 6C B0 BB EB 82

TVLA – dual output

1. T-value trace set

Automatically overlapped for all test vector data sets

2. Evolution plot



Lines: 1 Traces: 3 ☑ Overlap

~PinataHWAES + T-Zero-Multitest.trs
*1E-1
T-value Random vs MixCol_R5 (Traces: 166906_vs_166340)
T-value Random vs SubBytes_R5 (Traces: 166906_vs_166754)
T-value Threshold line
45 40 35 30 25 20 15 10 5
0 20 40 60 80 100 120 140 160 180 200 220 240 260 280
ns
~PinataHWAES + T-Zero-Multitest Evolution.trs
*1E-1
TVLA E-Plot Threshold line
TVLA E-Plot MixCol_R5
TVLA E-Plot SubBytes_R5
45 40 35 30 25 20 15 10 5 0
0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160
k# traces semi-constant set

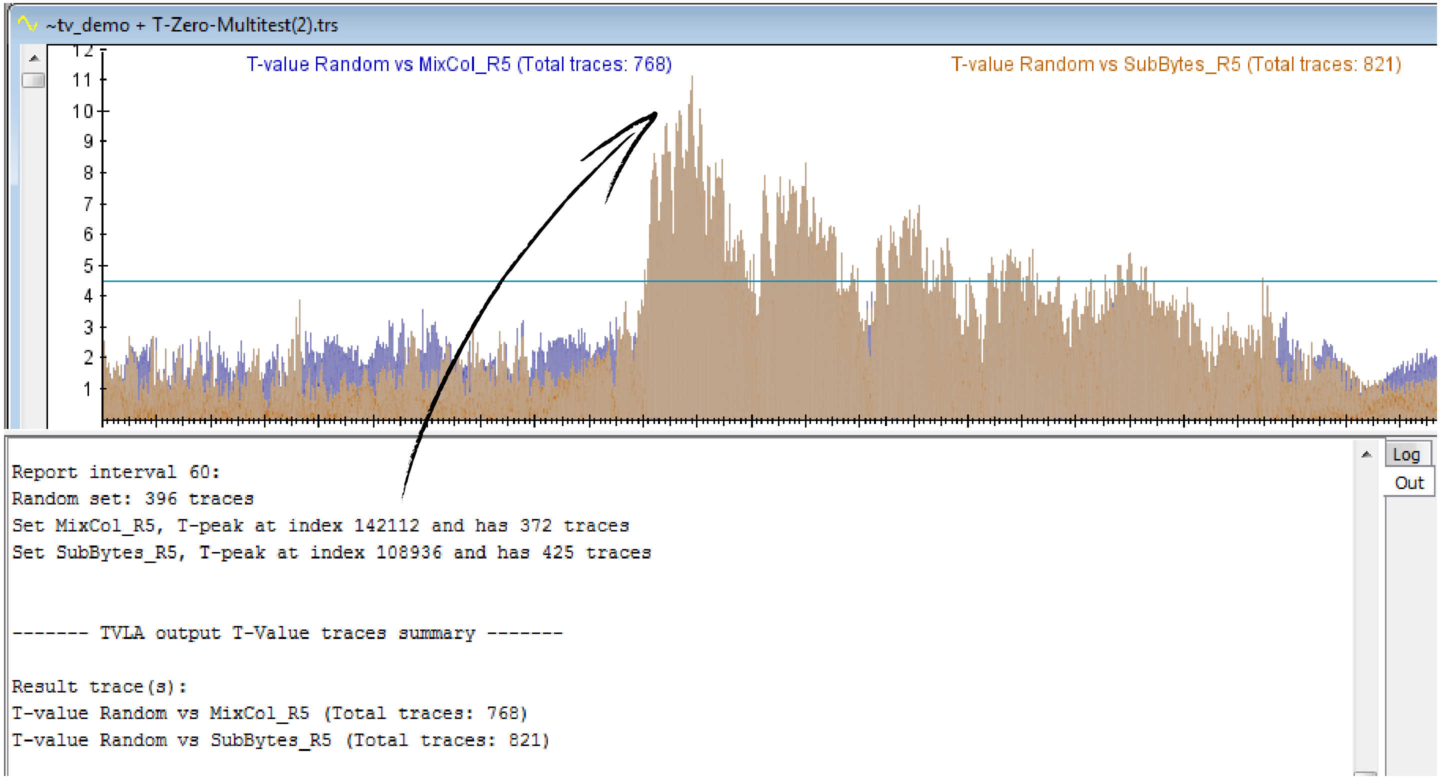
Threshold value 4.5

More leakage in SubBytes than MixColums

# traces in semi-contstant set

TVLA – cross check and conclude

Verify for several report intervals if chosen T-value peaks are not
ghost peaks (user interpretation):


~tv_demo + T-Zero-Multitest(2).trs
T-value Random vs MixCol_R5 (Total traces: 768)
T-value Random vs SubBytes_R5 (Total traces: 821)
12
11
10
9
8
7
6
5
4
3
2
1
Log
Out

```
Report interval 60:
Random set: 396 traces
Set MixCol_R5, T-peak at index 142112 and has 372 traces
Set SubBytes_R5, T-peak at index 108936 and has 425 traces


------- TVLA output T-Value traces summary -------

Result trace(s):
T-value Random vs MixCol_R5 (Total traces: 768)
T-value Random vs SubBytes_R5 (Total traces: 821)
```

TA on DES Key Scheduling

New technique

Template Analysis on DES Key Scheduling is a new method which is pending publication by Mathias Wagner (it has been shared with small groups).

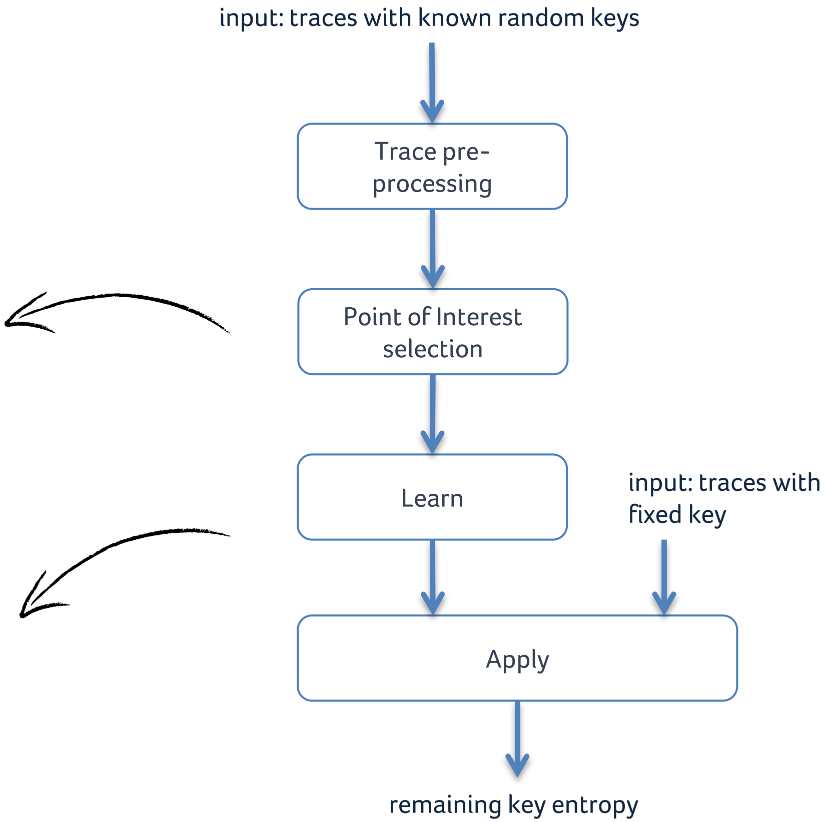
Features supported in Inspector SCA

Acquisition & POI selection:

- Known-key TA
- Target: leakage of XOR operations on key bits in the DES key scheduling function. The XOR operations are grouped together in four rings of 14 bits (A) and two rings of 28 bits (B) [pending publication]
- Rings: A & B applied, less sensitive to noise in POI selection than ring C

Learn & Apply:

- Probability models: Mean, +Var, +Cov with options Pooled and Centered
- Rings: Ring C applied, which combines Ring A and B and improves results for Template Analysis
- Template size: configurable.
- Note: when using +Cov the memory usage grows exponentially with the template bit size and number of POIs.



input: traces with known random keys

Trace pre-processing

Point of Interest selection

Learn

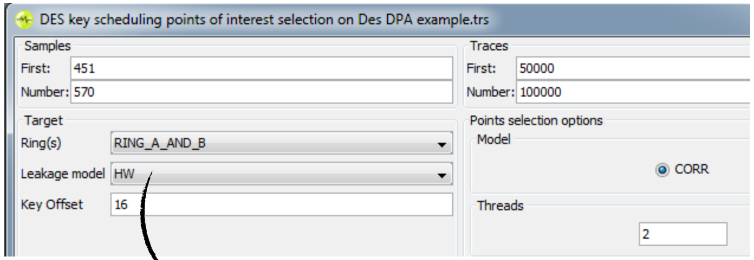
input: traces with fixed key

Apply

remaining key entropy

POI selection and learn phase

On the trace set with random known keys, perform POI selection:



DES key scheduling points of interest selection on Des DPA example.trs

Samples
First: 451
Number: 570

Traces
First: 50000
Number: 100000

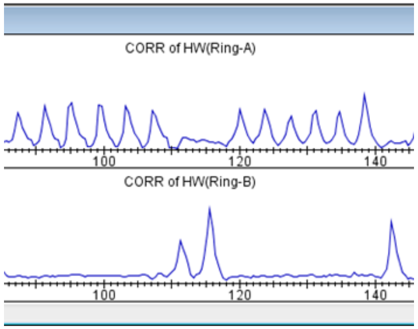
Target
Ring(s) RING_A_AND_B
Leakage model HW
Key Offset 16

Points selection options
Model
● CORR

Threads
2

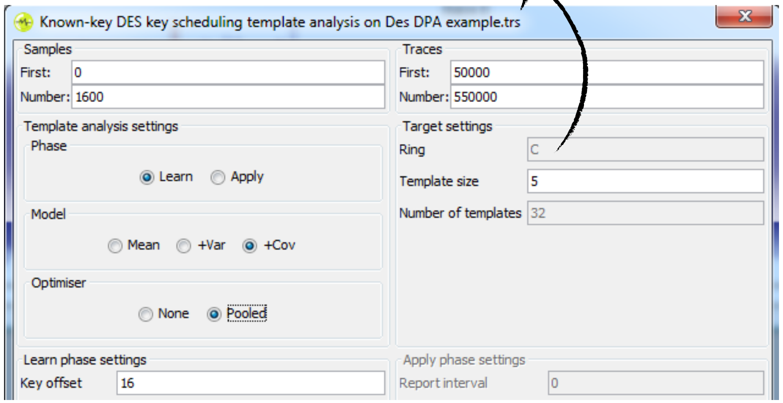
Ring A and B combined
is best for POI selection

Example POI selection results for Ring A and Ring B:



Learn phase:

Ring C is optimal for
Template Analysis



Known-key DES key scheduling template analysis on Des DPA example.trs

Samples
First: 0
Number: 1600

Traces
First: 50000
Number: 550000

Template analysis settings
Phase
● Learn ○ Apply
Model
○ Mean ○ +Var ● +Cov
Optimiser
○ None ● Pooled

Target settings
Ring C
Template size 5
Number of templates 32

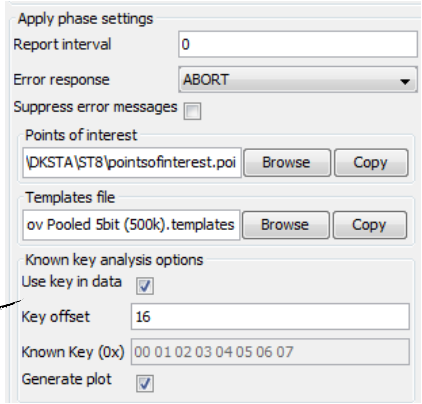
Learn phase settings
Key offset 16

Apply phase settings
Report interval 0

Key offset

Select template size

Apply phase:



Apply phase settings
Report interval 0
Error response ABORT
Suppress error messages □

Points of interest
\DKSTA\ST8\pointsofinterest.poi Browse Copy

Templates file
ov Pooled 5bit (500k).templates Browse Copy

Known key analysis options
Use key in data ☑
Key offset 16
Known Key (0x) 00 01 02 03 04 05 06 07
Generate plot ☑

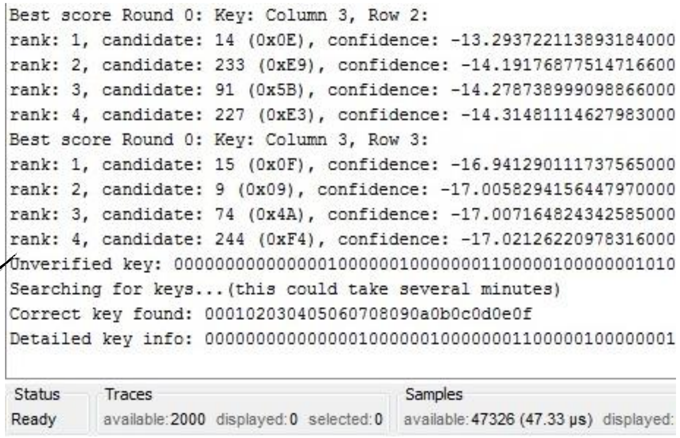
Option to use
key in data

Template Analysis – unknown key

Added unknown key analysis

Earlier this year several Template Analysis techniques for AES and DES were added to Inspector. Since then many users also asked us for the unknown key variant.

unknown key results
as you are used to it

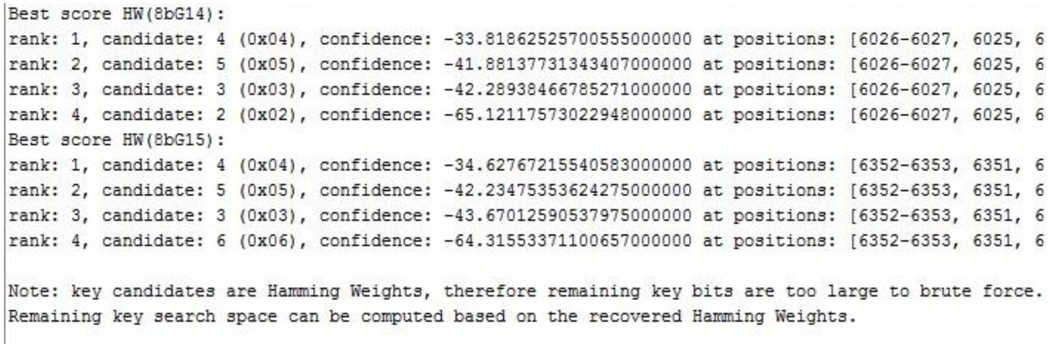


```
Best score Round 0: Key: Column 3, Row 2:
rank: 1, candidate: 14 (0x0E), confidence: -13.293722113893184000
rank: 2, candidate: 233 (0xE9), confidence: -14.191768775147166000
rank: 3, candidate: 91 (0x5B), confidence: -14.278738999098866000
rank: 4, candidate: 227 (0xE3), confidence: -14.314811146279830000
Best score Round 0: Key: Column 3, Row 3:
rank: 1, candidate: 15 (0x0F), confidence: -16.941290111737565000
rank: 2, candidate: 9 (0x09), confidence: -17.005829415644797000
rank: 3, candidate: 74 (0x4A), confidence: -17.007164824342585000
rank: 4, candidate: 244 (0xF4), confidence: -17.021262209783160000
Unverified key: 0000000000000001000000100000001100000100000001010
Searching for keys...(this could take several minutes)
Correct key found: 000102030405060708090a0b0c0d0e0f
Detailed key info: 0000000000000001000000100000001100000100000001
```

Status
Ready
Traces
available: 2000 displayed: 0 selected: 0
Samples
available: 47326 (47.33 µs) displayed:

Note: for hamming weight option in TA Key Loading, the unknown key output is different.

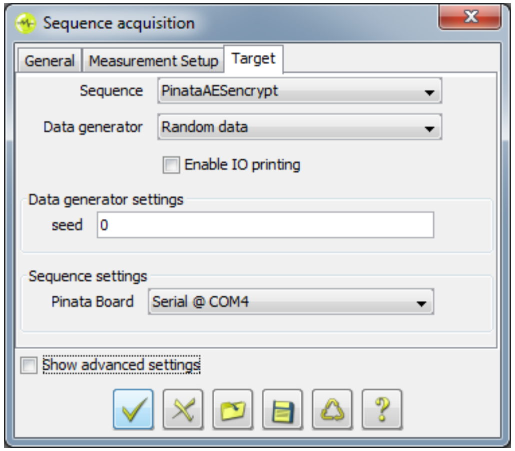
The candidates are hamming weights:



```
Best score HW(8bG14):
rank: 1, candidate: 4 (0x04), confidence: -33.818625257005550000000 at positions: [6026-6027, 6025, 6
rank: 2, candidate: 5 (0x05), confidence: -41.881377313434070000000 at positions: [6026-6027, 6025, 6
rank: 3, candidate: 3 (0x03), confidence: -42.289384667852710000000 at positions: [6026-6027, 6025, 6
rank: 4, candidate: 2 (0x02), confidence: -65.121175730229480000000 at positions: [6026-6027, 6025, 6
Best score HW(8bG15):
rank: 1, candidate: 4 (0x04), confidence: -34.627672155405830000000 at positions: [6352-6353, 6351, 6
rank: 2, candidate: 5 (0x05), confidence: -42.234753536242750000000 at positions: [6352-6353, 6351, 6
rank: 3, candidate: 3 (0x03), confidence: -43.670125905379750000000 at positions: [6352-6353, 6351, 6
rank: 4, candidate: 6 (0x06), confidence: -64.315533711006570000000 at positions: [6352-6353, 6351, 6

Note: key candidates are Hamming Weights, therefore remaining key bits are too large to brute force.
Remaining key search space can be computed based on the recovered Hamming Weights.
```

Data generator in Sequence

The data generator function is now also available in Sequence for embedded chip testing:



Sequence: PinataAESencrypt

Data generator: Random data

□ Enable IO printing

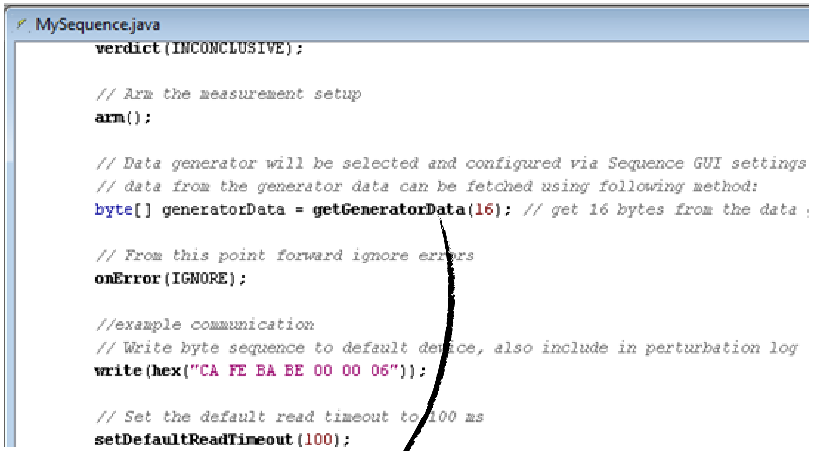
Data generator settings

seed: 0

Sequence settings

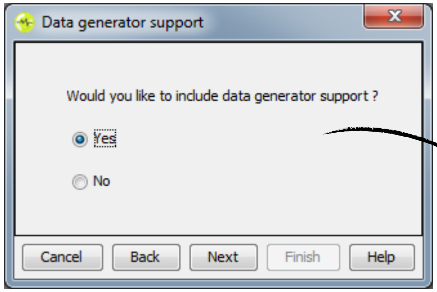
Pinata Board: Serial @ COM4

□ Show advanced settings



```
        verdict(INCONCLUSIVE);

        // Arm the measurement setup
        arm();

        // Data generator will be selected and configured via Sequence GUI settings
        // data from the generator data can be fetched using following method:
        byte[] generatorData = getGeneratorData(16); // get 16 bytes from the data

        // From this point forward ignore errors
        onError(IGNORE);

        //example communication
        // Write byte sequence to default device, also include in perturbation log
        write(hex("CA FE BA BE 00 00 06"));

        // Set the default read timeout to 100 ms
        setDefaultReadTimeout(100);
```

Method in Sequence for
data generator



Would you like to include data generator support ?

● Yes

○ No

When using the
Sequence wizard

Note : Pinata chip example modules were also changed to include the data generator in the GUI.

Miscellaneous

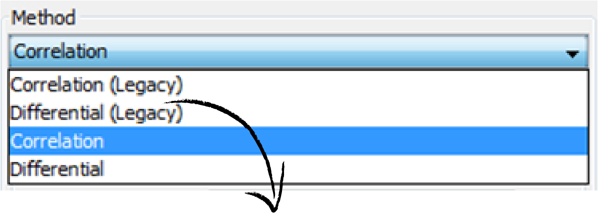
New Picoscope

Picoscope 3206D is now also supported. We deliver it with:

- Bandwidth 200 MHz
- Sample rate at 1 Gs/s
- Memory 512 MS
- 2 Channels

Multi-threaded correlation

Introduced successfully in 4.10, the legacy method was now ready for removal.



Method

Correlation

Correlation (Legacy)
Differential (Legacy)
Correlation
Differential

Removed legacy
single-threaded

Faster power control DPSS

Changing the power level of the DPSS laser is much faster with a new DPSS attenuator controlled from Inspector.

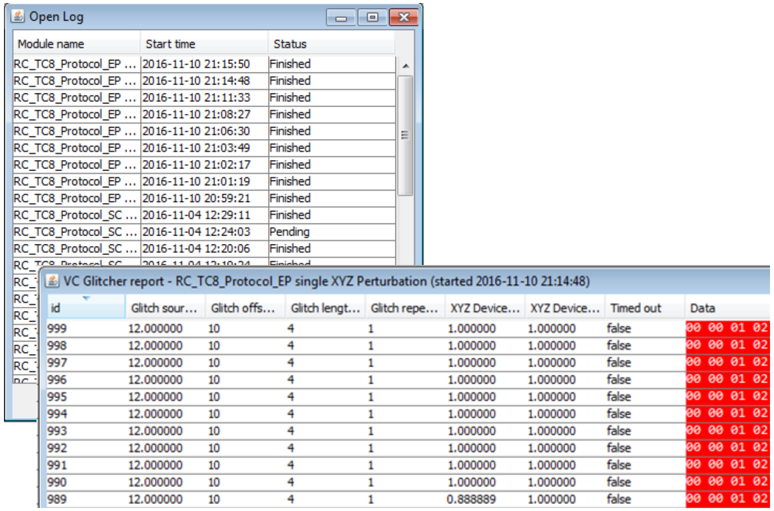
Crashes with “disk compression”

Hard disks that have “disk compression” enabled cause problems with large trace set acquisition and processing. This is a low level Windows issue and not related to Inspector software. An instruction not to enable disk compression has been added to the manual.

Miscellaneous

Perturbation log panel

The perturbation log history panel is now kept open when inspecting an individual perturbation record.



Open Log

Module name	Start time	Status
RC_TC8_Protocol_EP ...	2016-11-10 21:15:50	Finished
RC_TC8_Protocol_EP ...	2016-11-10 21:14:48	Finished
RC_TC8_Protocol_EP ...	2016-11-10 21:11:33	Finished
RC_TC8_Protocol_EP ...	2016-11-10 21:08:27	Finished
RC_TC8_Protocol_EP ...	2016-11-10 21:06:30	Finished
RC_TC8_Protocol_EP ...	2016-11-10 21:03:49	Finished
RC_TC8_Protocol_EP ...	2016-11-10 21:02:17	Finished
RC_TC8_Protocol_EP ...	2016-11-10 21:01:19	Finished
RC_TC8_Protocol_EP ...	2016-11-10 20:59:21	Finished
RC_TC8_Protocol_SC ...	2016-11-04 12:29:11	Finished
RC_TC8_Protocol_SC ...	2016-11-04 12:24:03	Pending
RC_TC8_Protocol_SC ...	2016-11-04 12:20:06	Finished

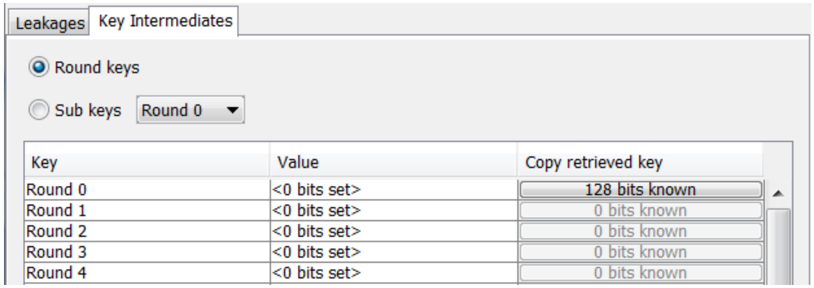
VC Glitcher report - RC_TC8_Protocol_EP single XYZ Perturbation (started 2016-11-10 21:14:48)

id	Glitch sour...	Glitch offs...	Glitch lengt...	Glitch repe...	XYZ Device...	XYZ Device...	Timed out	Data
999	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
998	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
997	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
996	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
995	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
994	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
993	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
992	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
991	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
990	12.000000	10	4	1	1.000000	1.000000	false	00 00 01 02
989	12.000000	10	4	1	0.888889	1.000000	false	00 00 01 02

Remembering key previous round

In first order analysis if you forgot to copy across the retrieved key, you would loose this key when running the next round. And, you would have to re-run the complete analysis!

Now the retrieved key is kept in memory so you can still copy it across.



● Round keys

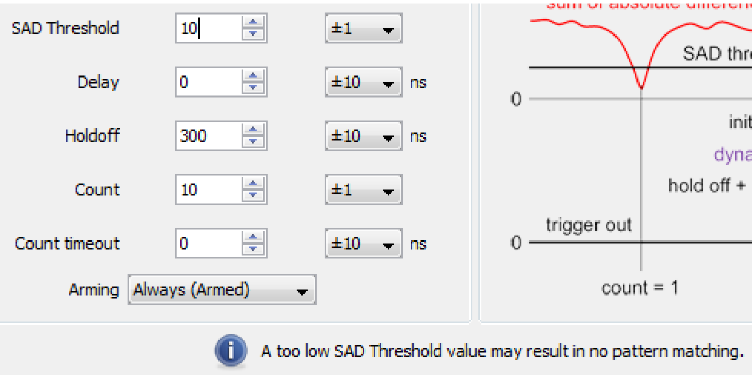
○ Sub keys   Round 0

Key	Value	Copy retrieved key
Round 0	<0 bits set>	128 bits known
Round 1	<0 bits set>	0 bits known
Round 2	<0 bits set>	0 bits known
Round 3	<0 bits set>	0 bits known
Round 4	<0 bits set>	0 bits known

Miscellaneous

icWaves parameters

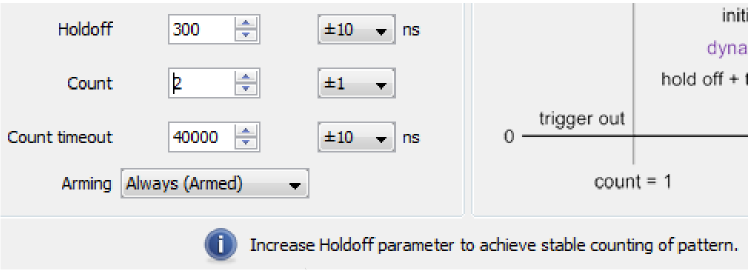
When counting multiple patterns, several parameters need the right combined configuration. Several suggestions are dynamically given to help in this, for example:



SAD Threshold	10	±1	
Delay	0	±10	ns
Holdoff	300	±10	ns
Count	10	±1	
Count timeout	0	±10	ns
Arming	Always (Armed)		

A too low SAD Threshold value may result in no pattern matching.

tip for SAD threshold



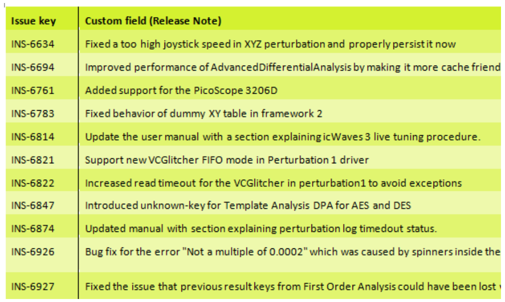
Holdoff	300	±10	ns
Count	2	±1	
Count timeout	40000	±10	ns
Arming	Always (Armed)		

Increase Holdoff parameter to achieve stable counting of pattern.

tip for Holdoff

Release notes & bug fixes

For the full list of bug fixes, please refer to the release notes:
https://www.riscure.com/security-tools/inspector-sca/#support



Issue key	Custom field (Release Note)
INS-6634	Fixed a too high joystick speed in XYZ perturbation and properly persist it now
INS-6694	Improved performance of AdvancedDifferentialAnalysis by making it more cache friend
INS-6761	Added support for the PicoScope 3206D
INS-6783	Fixed behavior of dummy XY table in framework 2
INS-6814	Update the user manual with a section explaining icWaves 3 live tuning procedure.
INS-6821	Support new VCGlitcher FIFO mode in Perturbation 1 driver
INS-6822	Increased read timeout for the VCGlitcher in perturbation1 to avoid exceptions
INS-6847	Introduced unknown-key for Template Analysis DPA for AES and DES
INS-6874	Updated manual with section explaining perturbation log timedout status.
INS-6926	Bug fix for the error "Not a multiple of 0.0002" which was caused by spinners inside the
INS-6927	Fixed the issue that previous result keys from First Order Analysis could have been lost

Upgrade procedure & SDK changes

Inspector 4.11 installation

Where

- Customers with Support Contract receive download link
- Download from Riscure download portal

Installation guidance

- Inspector software can be installed on the same PC workstation next to your previous version. You can still revert back to the previous version if you want to.
- API is backwards compatible with Inspector 4.7 and onwards.

Your own modules & traces

- Inspector software points by default to the same user module folder as previous versions.
- Your own modules and traces from Inspector 4.7 and onwards are compatible with this Inspector release.
- In case you have trouble porting an older module to this Inspector version, please contact our support portal for assistance.

SDK and firmware updates

- None

Please contact Riscure for more information

You can reach us by email : inforequest@riscure.com

by phone : +31 15 251 4090 US: +1 650 646 9979

Or on the web: riscure.com

riscure