First-Order DPA Attack Against AES in Counter Mode w/ Unknown Counter

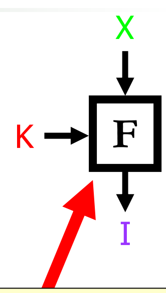
Josh Jaffe
CHES 2007

Cryptography Research, Inc.
www.cryptography.com
575 Market St., 21st Floor, San Francisco, CA 94105

CRYPTOGRAPHY RESEARCH

DPA Attack, typical structure

- Collect a set of power traces, with corresponding known variable data, X.
- Guess the value of a constant K that is mixed with X.
- Predict a variable intermediate I and use values to partition the set of traces.
- DPA test: compute differences between average of each partition. (Compute significance of differences in power measurements between subsets.)



"F" is a small portion of the cipher that the attack chooses to focus on

Result: If intermediate I leaks, the DPA test shows spikes when guess K is correct.

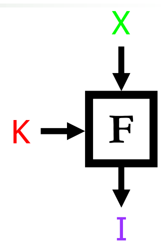
CRYPTOGRAPHY RESEARCH

CRYPTOGRAPHY RESEARCH

DPA Attack, typical structure

Assumptions (typical attack):

- Input or output X to block cipher is known: "Attack input" data.
- Attack input data X varies significantly over set of traces analyzed.
  - usually either chosen or random input



Counter Mode

- Counter mode uses a block cipher B in a stream cipher mode:
  - O_T = B_enc(C + T, K)
  - $Y_T = O_T \oplus X_T$

Initial Counter
Key
Ciphertext
Plaintext

- Differences from traditional DPA attack:
  - Assume inputs (counter values) and outputs of block cipher are not known
  - Only a few bits of the input data (the counter value) vary over set of collected traces.

Results

- The new attack develops a method for handling unknown input (counter values) and output.
- The new attack develops a method for handling protocols in which only a few bits of the input are varying.
- Contrary to what I've previously said, a high-order attack is not necessary here.

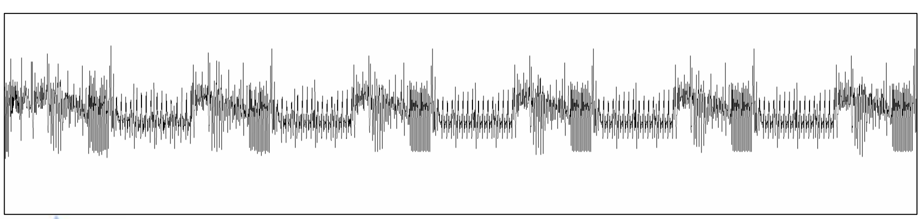
Attack Details

Overview of the attack

- Collect Data (e.g. 2^{16} consecutive traces).
- Use DPA to attack the two low-order bytes of the counter (first round).
- Propagate attack into AES rounds two and three.
- The input to round 4 is fully known, fully variable.
  - Apply standard DPA to recover round key

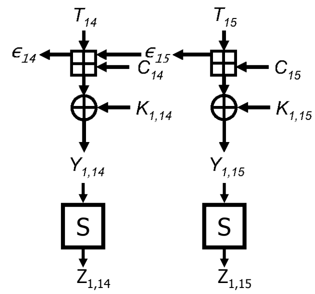
Data Collection

- Collect a few more than the number of traces needed to mount a 1st-order DPA attack on the device.
  - As implemented: 2^{16} traces. (2h37m)
- Record data for first 4 or 5 rounds of each encryption



Recovering two low-order bytes in Round 1

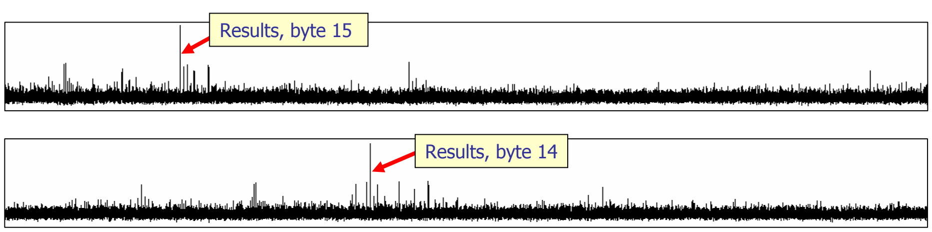
- Input values
  - Counter (AES input) is $C_T = C + T$.
  - T is a known value, starting at 0.
  - Treat T as the input to the cipher.



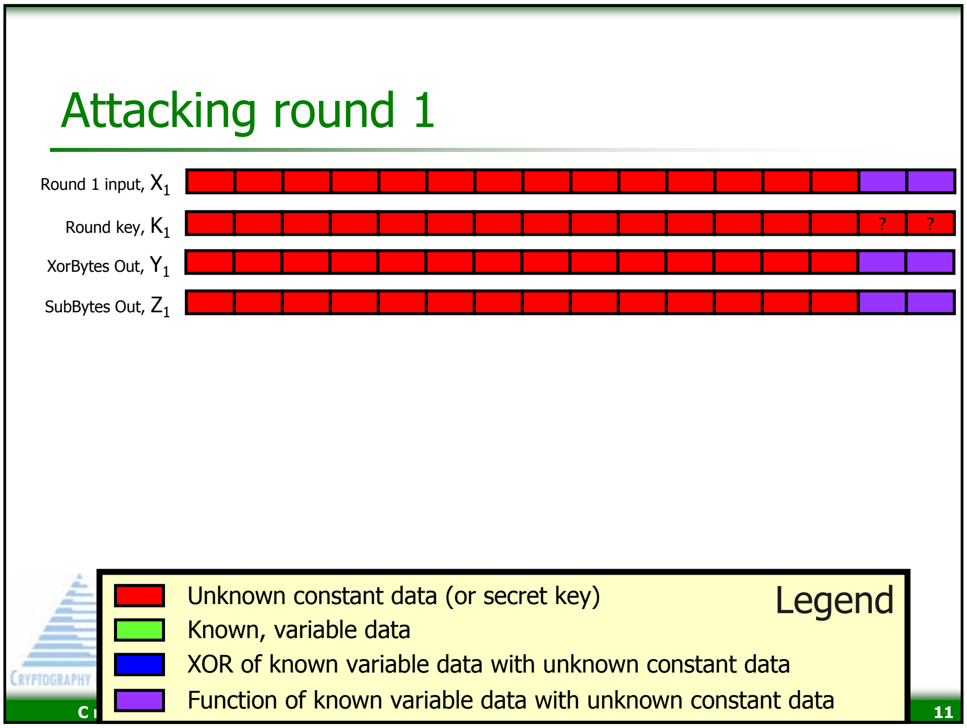
- DPA attack:
  - Guess C_{15} and $K_{1,15}$. Predict SubBytes output byte $Z_{1,15}$.
  - Repeat attack to recover C_{14}, $K_{1,14}$, and ϵ_{15}.
  - Attack does not actually recover ϵ_{14}.
    - For remaining stages of the attack, work only with the subset of traces for which bytes 0 through 13 of C_T are constant.
    - See paper for details

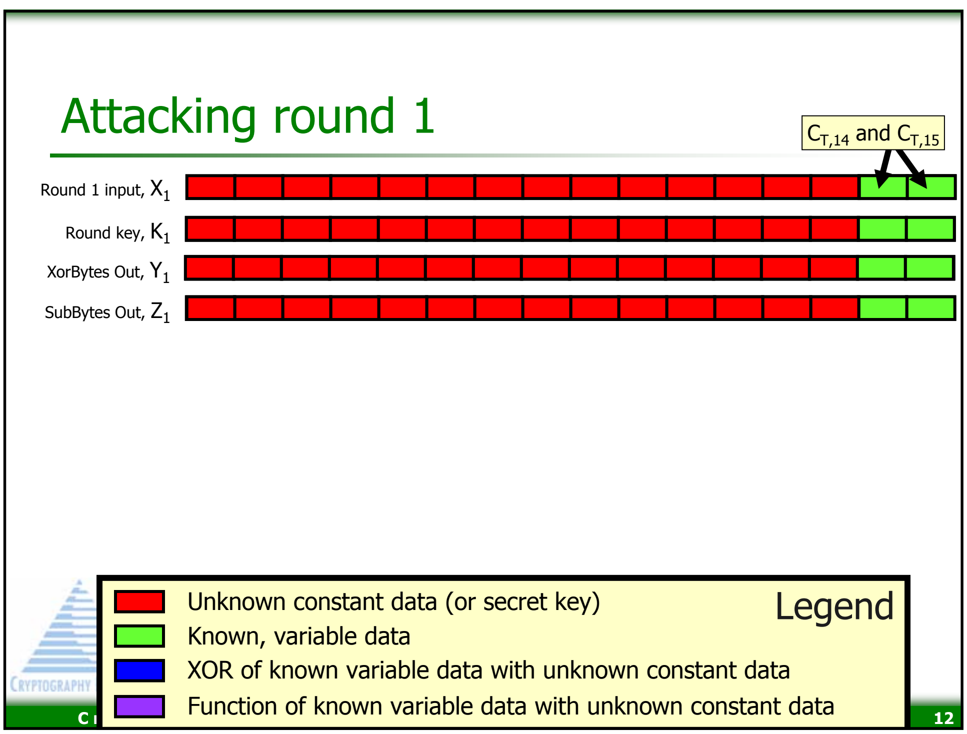
Results (power traces)

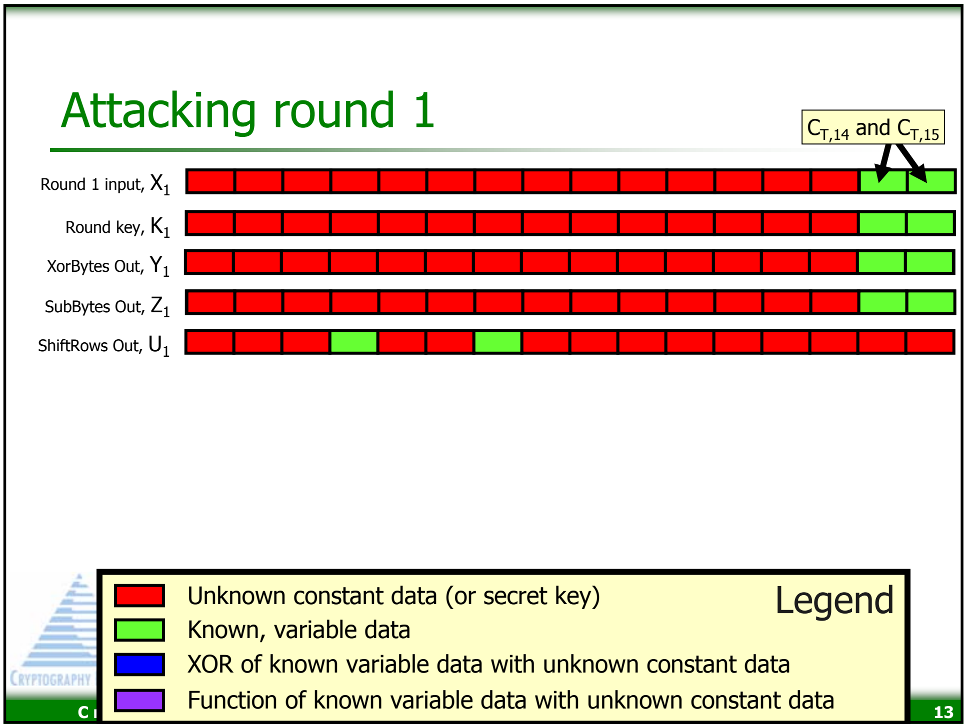
- Attack implementation / results:
  - $K_{1,15,lo}$ = 30h, $C_{15,lo}$ = 42h, b_{15} = 0.
  - $K_{1,14,lo}$ = 65h, $C_{14,lo}$ = 35h, b_{14} = 0, and $C_{15,hi}$ = 0 so $K_{1,15,hi}$ = 0.

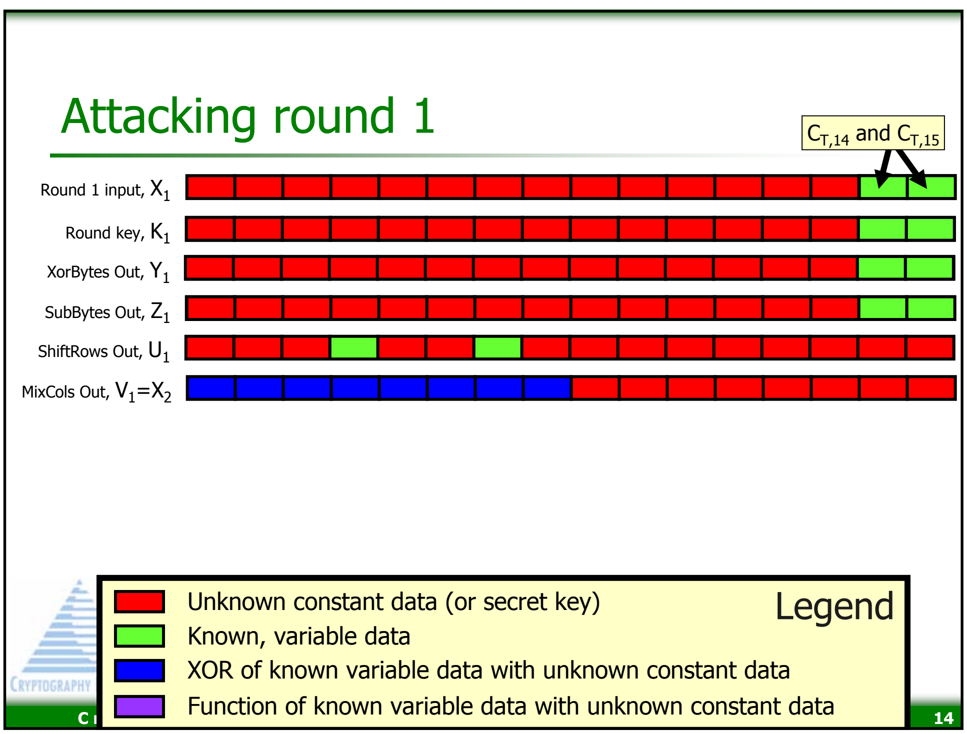


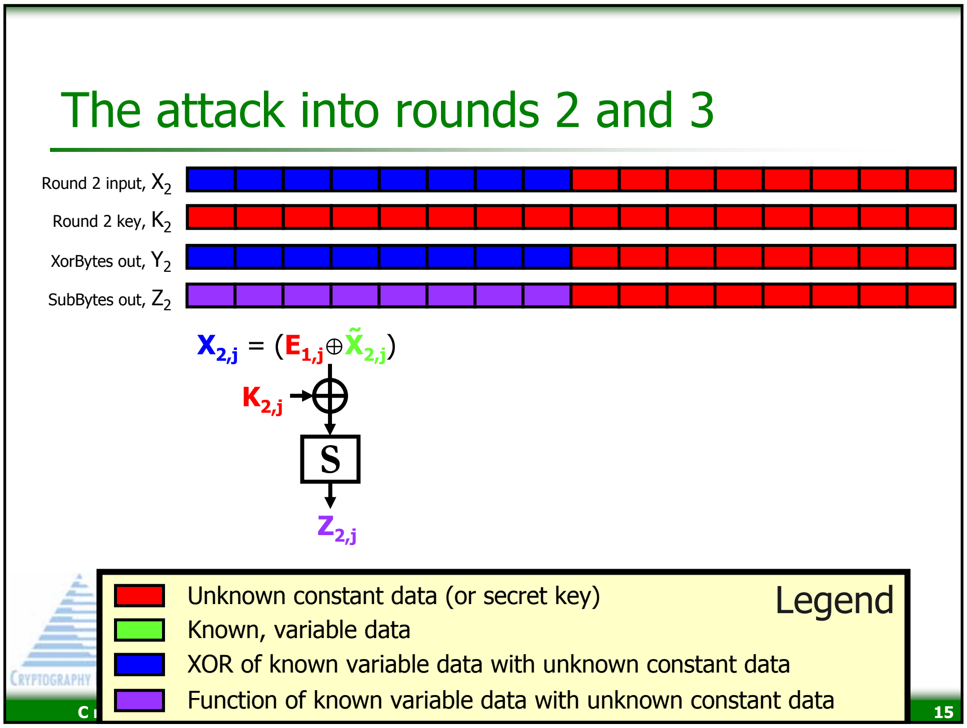
- X axis: guess of K and C.
- Y axis: amplitude of spikes observed (absolute value).

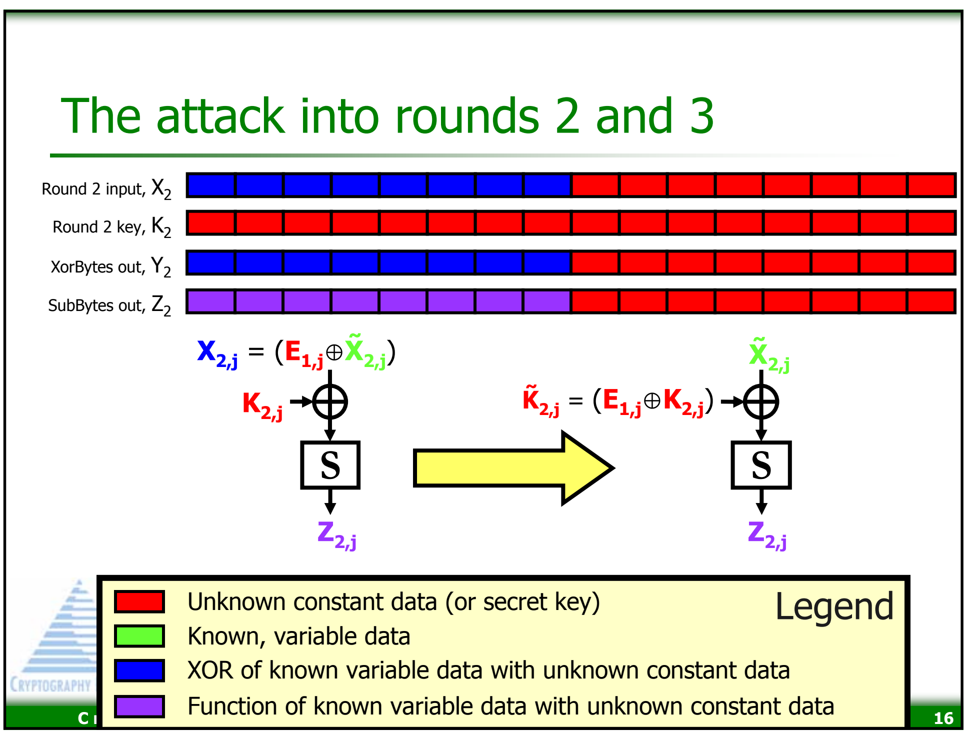

Attacking round 1
Round 1 input, X_1
Round key, K_1
?
?
XorBytes Out, Y_1
SubBytes Out, Z_1
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
11

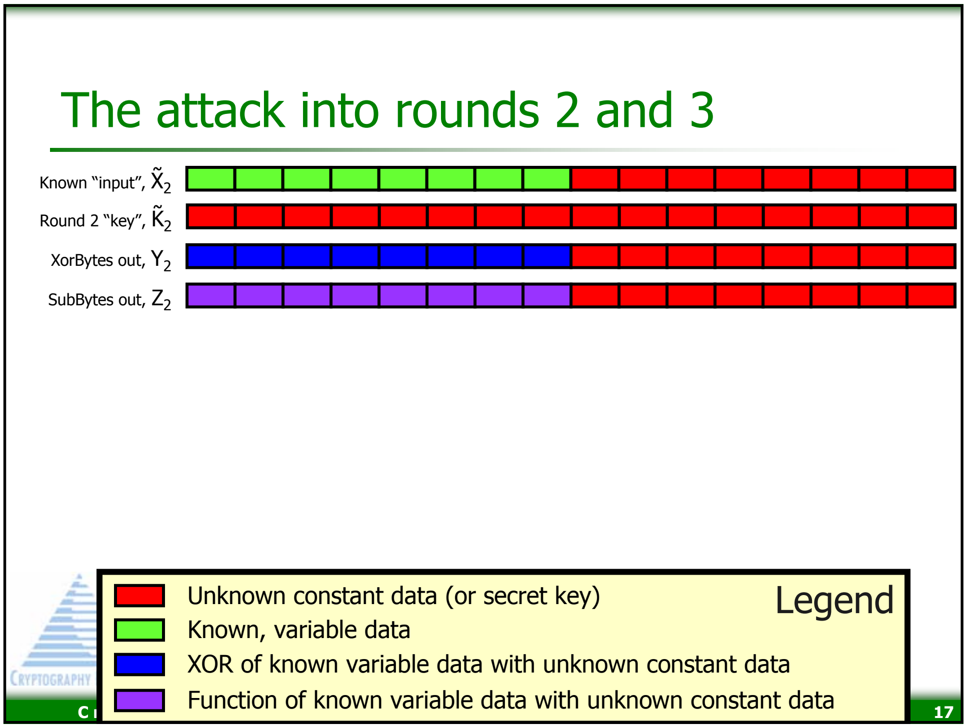

Attacking round 1
$C_{T,14}$ and $C_{T,15}$
Round 1 input, X_1
Round key, K_1
XorBytes Out, Y_1
SubBytes Out, Z_1
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
12

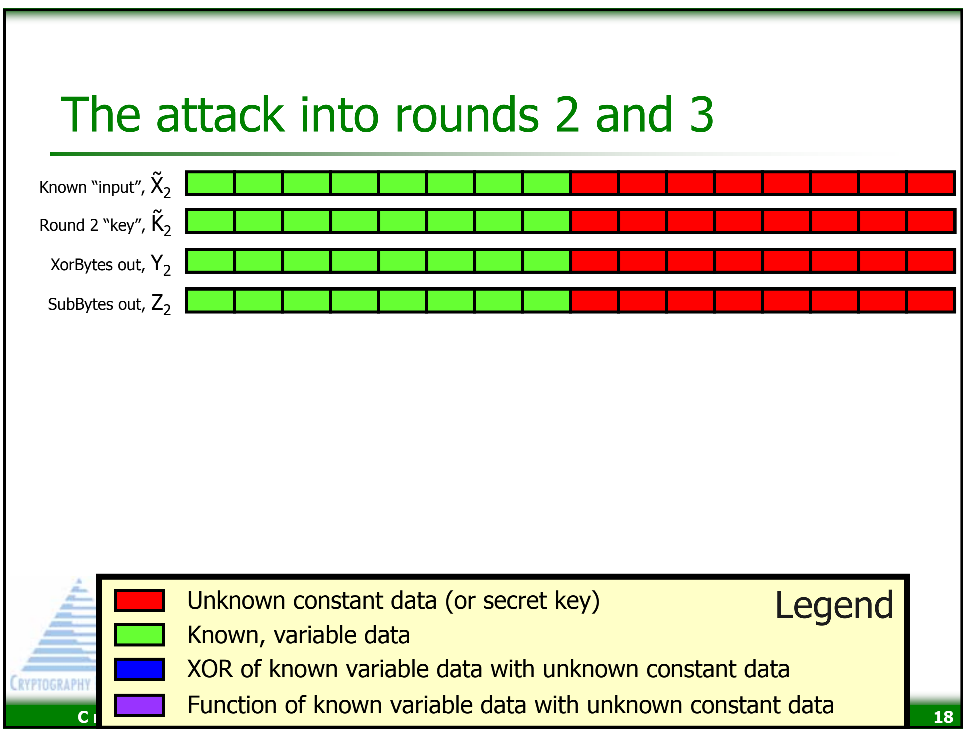

Attacking round 1
C_{T,14} and C_{T,15}
Round 1 input, X_1
Round key, K_1
XorBytes Out, Y_1
SubBytes Out, Z_1
ShiftRows Out, U_1
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
13

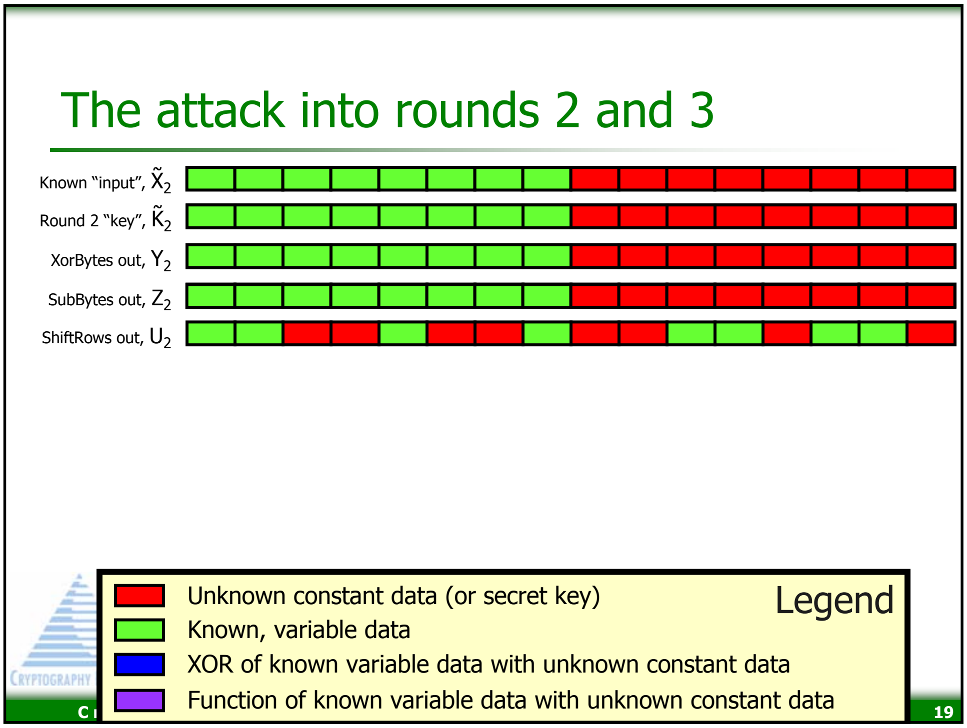

Attacking round 1
C_{T,14} and C_{T,15}
Round 1 input, X_1
Round key, K_1
XorBytes Out, Y_1
SubBytes Out, Z_1
ShiftRows Out, U_1
MixCols Out, V_1=X_2
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
14

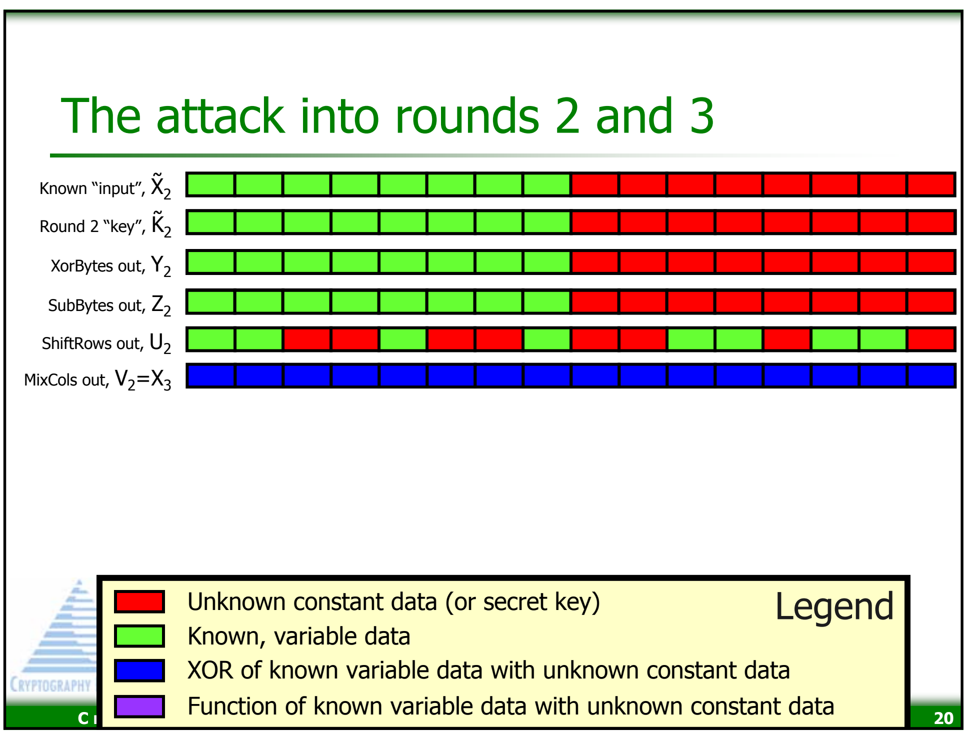

The attack into rounds 2 and 3
Round 2 input, X_2
Round 2 key, K_2
XorBytes out, Y_2
SubBytes out, Z_2
$X_{2,j} = (E_{1,j} \oplus \tilde{X}_{2,j})$
$K_{2,j}$
S
$Z_{2,j}$
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
15

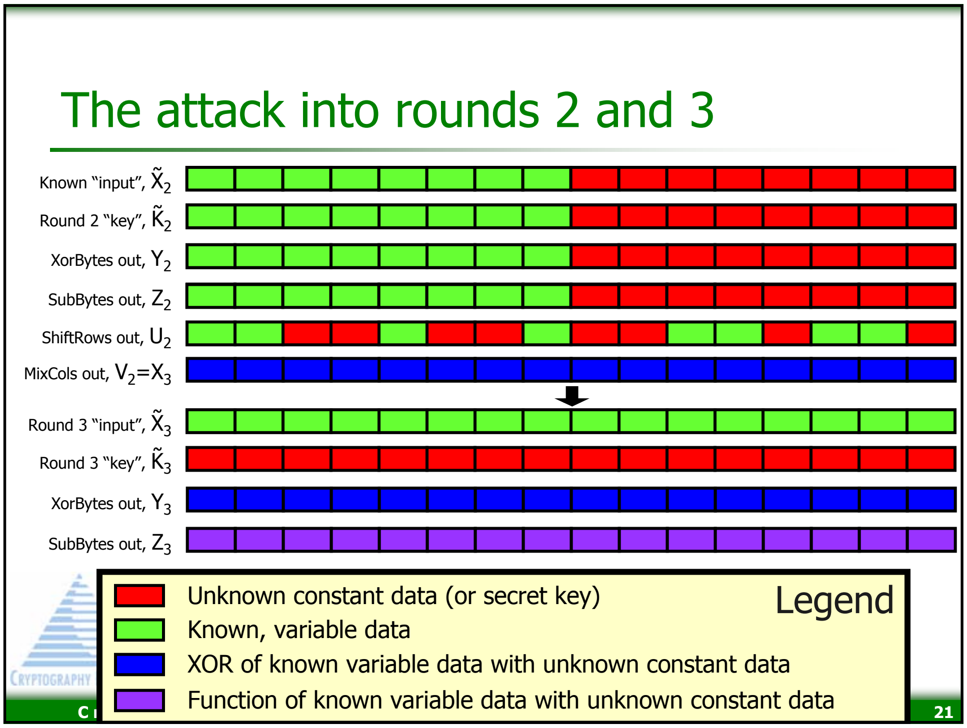

The attack into rounds 2 and 3
Round 2 input, X_2
Round 2 key, K_2
XorBytes out, Y_2
SubBytes out, Z_2
$X_{2,j} = (E_{1,j} \oplus \tilde{X}_{2,j})$
$K_{2,j}$
S
$Z_{2,j}$
$\tilde{X}_{2,j}$
$\tilde{K}_{2,j} = (E_{1,j} \oplus K_{2,j})$
S
$Z_{2,j}$
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
16

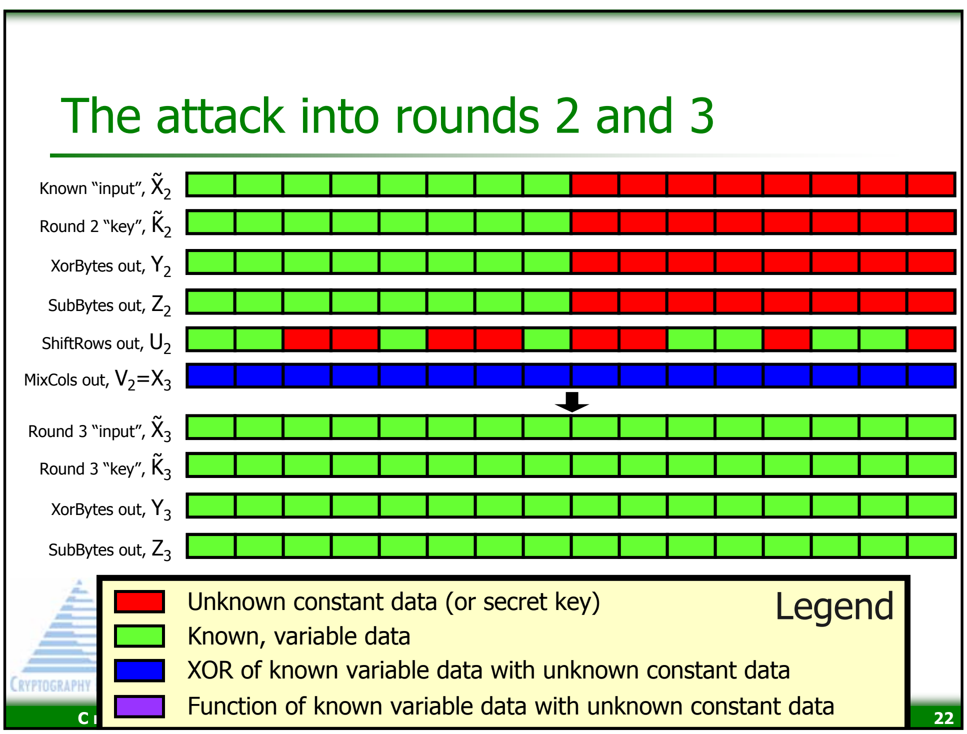

The attack into rounds 2 and 3
Known "input", $\tilde{X}_2$
Round 2 "key", $\tilde{K}_2$
XorBytes out, Y_2
SubBytes out, Z_2
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
17


The attack into rounds 2 and 3
Known "input", $\tilde{X}_2$
Round 2 "key", $\tilde{K}_2$
XorBytes out, Y_2
SubBytes out, Z_2
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
18

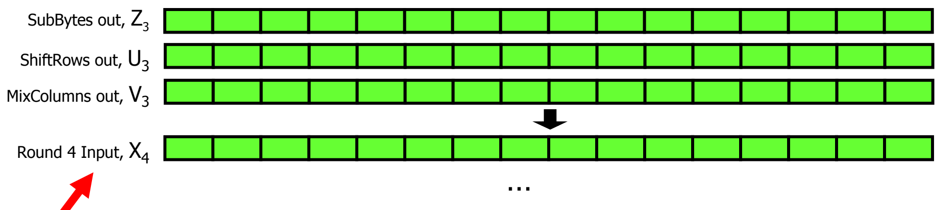

The attack into rounds 2 and 3
Known "input", $\tilde{X}_2$
Round 2 "key", $\tilde{K}_2$
XorBytes out, Y_2
SubBytes out, Z_2
ShiftRows out, U_2
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
19


The attack into rounds 2 and 3
Known "input", $\tilde{X}_2$
Round 2 "key", $\tilde{K}_2$
XorBytes out, Y_2
SubBytes out, Z_2
ShiftRows out, U_2
MixCols out, $V_2=X_3$
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
20


The attack into rounds 2 and 3
Known "input", $\tilde{X}_2$
Round 2 "key", $\tilde{K}_2$
XorBytes out, Y_2
SubBytes out, Z_2
ShiftRows out, U_2
MixCols out, $V_2=X_3$
Round 3 "input", $\tilde{X}_3$
Round 3 "key", $\tilde{K}_3$
XorBytes out, Y_3
SubBytes out, Z_3
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
21


The attack into rounds 2 and 3
Known "input", $\tilde{X}_2$
Round 2 "key", $\tilde{K}_2$
XorBytes out, Y_2
SubBytes out, Z_2
ShiftRows out, U_2
MixCols out, $V_2=X_3$
Round 3 "input", $\tilde{X}_3$
Round 3 "key", $\tilde{K}_3$
XorBytes out, Y_3
SubBytes out, Z_3
Legend
Unknown constant data (or secret key)
Known, variable data
XOR of known variable data with unknown constant data
Function of known variable data with unknown constant data
22

The attack into round 4



At this point the attack has overcome both challenges presented by counter mode:

- Input (to round 4) is fully known
- Input is varying randomly.

The attack continues from start of round 4 using standard DPA methods.

Legend

- Unknown constant data (or secret key)
- Known, variable data
- XOR of known variable data with unknown constant data
- Function of known variable data with unknown constant data

Conclusions (1/3)

- A 1st-order DPA attack can be possible if input is not known but relationship between inputs can be expressed in terms of known values.
  - Attack works in realistic, low-leakage-rate DPA scenarios.
  - No reliance on SPA high-amplitude leaks.

Conclusions (2/3)

- The general method for dealing with constant portions of input message data: Ignore it until it is mixed with variable data.
  - A. If mixing function is nonlinear:
    - Can attack now with DPA. (e.g. SubBytes of XorBytes.)
    - Can defer attack, but increase size of subsequent key search exponentially.
  - B. If mixing function is linear:
    - Can postpone the attack until later.

Conclusions (3/3)

- In some analysis scenarios, attacking counter mode will be easier.
  - For example, analysis lab need not implement complicated interface to device under test.
  - Only need to measure & record power consumption.
- Summary: Just because a device is in counter mode, with secret counter value, don't assume that the device is secure!

End

Questions

- Why 2^{16} traces? Isn't that a lot?
  - DPAWS is fast… crunches the numbers in 4 minutes.
  - Can crunch 5k traces in 4 seconds once they're in the cache.
- What if leakage rate is so low that (say) 200k traces are needed?
  - More unknown bytes are varying…
  - Attack may still be possible with modifications.
- What if total data is less than 2^{16} blocks / counter?
  - E.g. Ethernet frame size: ~4k packets
  - Attack will succeed/fail depending on leakage rate of device.
- What if counter update is LFSR, not increment?
  - Attack the LFSR construction – it's a different attack.
- What about "bit permuting" ciphers (DES) / or those who update noncontiguous bits of counter?
  - Either counter update is localized (most S-box inputs constant) or is not localized (most S-box inputs are variable).
  - In the latter case, there may be no need to push the attack all the way into the fourth round.