硬體安全HW1

第四組

1. Code:

"""This script is an example of a timing attack on a simple password checker.

It is the result of Tutorial B3-1 from the ChipWhisperer Wiki.

"""

import chipwhisperer as cw

from chipwhisperer.capture.auxiliary.ResetCW1173Read import ResetCW1173

# GUI compatibility

try:

scope = self.scope

target = self.target

aux\_list = self.aux\_list

except NameError:

pass

# Set up scope

scope.gain.gain = 45

scope.adc.samples = 2000

scope.adc.offset = 0

scope.adc.basic\_mode = "rising\_edge"

scope.clock.clkgen\_freq = 7370000

scope.clock.adc\_src = "clkgen\_x4"

scope.trigger.triggers = "tio4"

scope.io.tio1 = "serial\_rx"

scope.io.tio2 = "serial\_tx"

scope.io.hs2 = "clkgen"

# Set up target

target.key\_cmd = ""

target.go\_cmd = "h0px3\n"

target.output\_cmd = ""

# Set up aux module to reset target before capture

resetter = ResetCW1173(xmega=True, delay\_ms=1200)

aux\_list.register(resetter.resetThenDelay, "before\_trace")

# Test one capture

cw.captureN(self.scope, self.target, None, self.aux\_list, self.ktp, 1)

trace = scope.getLastTrace()

print trace

# Crack the first letter

password = ''

trylist = 'abcdefghijklmnopqrstuvwxyz0123456789'

for i in range(5):

for c in trylist:

# Get a power trace using our next attempt

nextPass = password + '{}'.format(c) + "\n"

target.go\_cmd = nextPass

cw.captureN(self.scope, self.target, None, self.aux\_list, self.ktp, 1)

# Grab the trace

nextTrace = scope.getLastTrace()

# Check location 229, 269, etc. If it's too low, we've failed

if nextTrace[289 + 40\*i] < -0.25 and nextTrace[289 + 40\*i] > -0.4:

continue

# If we got here, we've found the right letter

password += c

print '{} characters: {}'.format(i+1, password)

break

1. 過程

一張含有 螢幕擷取畫面, 電腦, 室內, 膝上型電腦 的圖片

自動產生的描述一張含有 螢幕擷取畫面, 電腦, 室內, 膝上型電腦 的圖片

自動產生的描述一張含有 螢幕擷取畫面, 室內, 電腦, 膝上型電腦 的圖片

自動產生的描述

以上三圖分別為aaaaa, haaaa, h0px3的power trace

在aaaaa的power trace裡，可以發現在sample 289有個尖峰；則在haaaa的power trace裡，可以發現那個尖峰移到329了，而該尖峰值略小於-0.25

因此將判斷條件改為下列參數

if nextTrace[289 + 40\*i] < -0.25

然而這個條件只能找到第四個，也就是h0px，第五個則無法順利找到

經過仔細觀察發現，第五個要檢查的尖峰位置在289+40\*4 = 449，而當全對，也就是h0px3時，雖然第五個數字對了，但power trace 449的位置會比-0.25更低，略低於-0.4，因此條件修改為以下

if nextTrace[289 + 40\*i] < -0.25 and nextTrace[289 + 40\*i] > -0.4:

1. Result

一張含有 螢幕擷取畫面, 室內, 電腦, 膝上型電腦 的圖片

自動產生的描述