

Untitled


BY: A GUEST ON SEP 3RD, 2015 | SYNTAX: PYTHON | SIZE: 2.76 KB | VIEWS: 77 | EXPIRES: NEVER

DOWNLOAD | RAW | EMBED | REPORT ABUSE | PRINT | QR CODE | CLONE

OpenVZ or KVM

Windows / CentOS / Ubuntu

Premium hardware / Good privacy

 VPS.ag
PRIVATE SERVERS

Get your VPS now (as low as €10)

€10
STARTING FROM

TEXT BELOW IS SELECTED. PLEASE PRESS CTRL+C TO COPY TO YOUR CLIPBOARD. (⌘+C ON MAC)

```
1. #!/usr/bin/env python2.7
2.
3. from z3 import *
4.
5. k = [0] * 16
6. for i in range(16):
7.     k[i] = [ 0x11, 0xaa, 0x55, 0x33 ][i//4]
8.
9. b = [ BitVec('b%d' % i, 32) for i in range(16) ]
10. v = [ b[i] ^ k[i] for i in range(16) ]
11. s = Solver()
12.
13. for i in range(16):
14.     s.add(b[i] & 0xff == b[i])
15.
16.
17. # _ _ _ _
18. # / / / / _ _ _ /
19. # / / / / _ _ / _ \
20. # / _ / / _ \ _ \ / /
21. # / / / _ \ _ / _ / /
22. #
23.
24. S1 = BitVec('s1', 32)
25. S2 = BitVec('s2', 32)
26. S3 = BitVec('s3', 32)
27. S4 = BitVec('s4', 32)
28.
29. for i in range(16):
30.     val = v[i]
31.     s.add(33 <= val <= 122)
32.     S1 = S1 + val
33.     S2 = 37 * val + RotateRight(S2, 23)
34.     S3 = val ^ 8 * S3
35.     S4 = val + 32 * S4
36.
37. s.add(S1 == 1720)
38. s.add(S2 == 0xDCCE74)
39. s.add(S3 == 0xEBAF1446)
40. s.add(S4 == 0x9961270E)
41.
42.
43.
44. # _ _ _ _ _
45. # / _ / _ _ _ / ( ) / ( ) _ _ _
46. # / / / _ \ _ \ _ \ / / / / _ \ _ \
47. # / / / ( ) / / / / ( ) / / / ( ) / / /
48. # \ _ \ / / / / \ _ \ / / \ / \ / \ / /
49. #
50.
51.
52. # Just copy & paste code from IDA
53.
54. v12 = b[1] ^ b[0]
55. v13 = b[2] ^ b[1] ^ b[0]
56. v14 = b[3] ^ b[2] ^ b[1] ^ b[0]
57. v15 = b[4] ^ b[3] ^ b[2] ^ b[1] ^ b[0]
```

0

Public P

0

Untit
8 sec

Untit
9 sec

Untit
13 se

Untit
22 se

wut
22 se

Untit
26 se

Untit
43 se

Untit
48 se

Up

Pas

a

U

m

fe

clie

lea

CH

```

58. v16 = b[5] ^ b[4] ^ b[3] ^ b[2] ^ b[1] ^ b[0]
59. v17 = b[6] ^ b[5] ^ b[4] ^ b[3] ^ b[2] ^ b[1] ^ b[0]
60. v18 = b[7] ^ b[6] ^ b[5] ^ b[4] ^ b[3] ^ b[2] ^ b[1] ^ b[0]
61. v19 = b[8] ^ b[7] ^ b[6] ^ b[5] ^ b[4] ^ b[3] ^ b[2] ^ b[1] ^ b[0]
62. v20 = b[9] ^ v19
63. v21 = b[10] ^ b[9] ^ v19
64. v22 = b[11] ^ b[10] ^ b[9] ^ v19
65. v23 = b[12] ^ b[11] ^ b[10] ^ b[9] ^ v19
66. v24 = b[14] ^ b[13] ^ b[12] ^ b[11] ^ b[10] ^ b[9] ^ v19
67. v25 = b[13] ^ b[12] ^ b[11] ^ b[10] ^ b[9] ^ v19
68. v26 = b[15] ^ b[14] ^ b[13] ^ b[12] ^ b[11] ^ b[10] ^ b[9] ^ v19
69.
70.
71. s.add(0 == 0xff & (b[0] ^ 0x63))
72. s.add(0 == 0xff & (((v12 << 1) | (v12 >> 1)) ^ 0x2F))
73. s.add(0 == 0xff & (((v13 << 2) | (v13 >> 2)) ^ 0xDC))
74. s.add(0 == 0xff & (((v14 << 3) | (v14 >> 3)) ^ 0x20))
75. s.add(0 == 0xff & (((v15 << 4) | (v15 >> 4)) ^ 0xCD))
76. s.add(0 == 0xff & (((v16 << 5) | (v16 >> 5)) ^ 0xA0))
77. s.add(0 == 0xff & (((v17 << 6) | (v17 >> 6)) ^ 0x83))
78. s.add(0 == 0xff & (((v18 << 7) | (v18 >> 7)))
79. s.add(0 == 0xff & (v19 ^ 0x30))
80. s.add(0 == 0xff & (((v20 << 1) | (v20 >> 1)) ^ 0x7D))
81. s.add(0 == 0xff & (((v21 << 2) | (v21 >> 2)) ^ 0x19))
82. s.add(0 == 0xff & (((v22 << 3) | (v22 >> 3)) ^ 4))
83. s.add(0 == 0xff & (((v23 << 4) | (v23 >> 4)) ^ 0xC4))
84. s.add(0 == 0xff & (((v25 << 5) | (v25 >> 5)) ^ 0x20))
85. s.add(0 == 0xff & (((v24 << 6) | (v24 >> 6)) ^ 0xC1))
86. s.add(0 == 0xff & (((v26 << 7) | (v26 >> 7)))
87.
88.
89. # You need to guess XD
90. #s.add(v[1] == ord('e'))
91. #s.add(v[2] == ord('v'))
92. #s.add(v[6] == ord('i'))
93.
94. print(s.check())
95. m = s.model()
96.
97. res = ""
98. for i in range(16):
99.     v = int(str(m[b[i]]))
100.    c = chr(v ^ k[i])
101.    print('%d: %d %s' % (i, v, c))
102.    res += c
103. print("Key is %s" % res)

```

RAW Paste Data

[clone this paste](#)

```

from z3 import *

k = [0] * 16
for i in range(16):
    k[i] = [ 0x11, 0xaa, 0x55, 0x33 ][i//4]

b = [ BitVec('b%d' % i, 32) for i in range(16) ]
v = [ b[i] ^ k[i] for i in range(16) ]
s = Solver()

for i in range(16):
    s.add(b[i] & 0xff == v[i])

```

Pastebin T-shirt Shop
high quality geeky t-shirts



