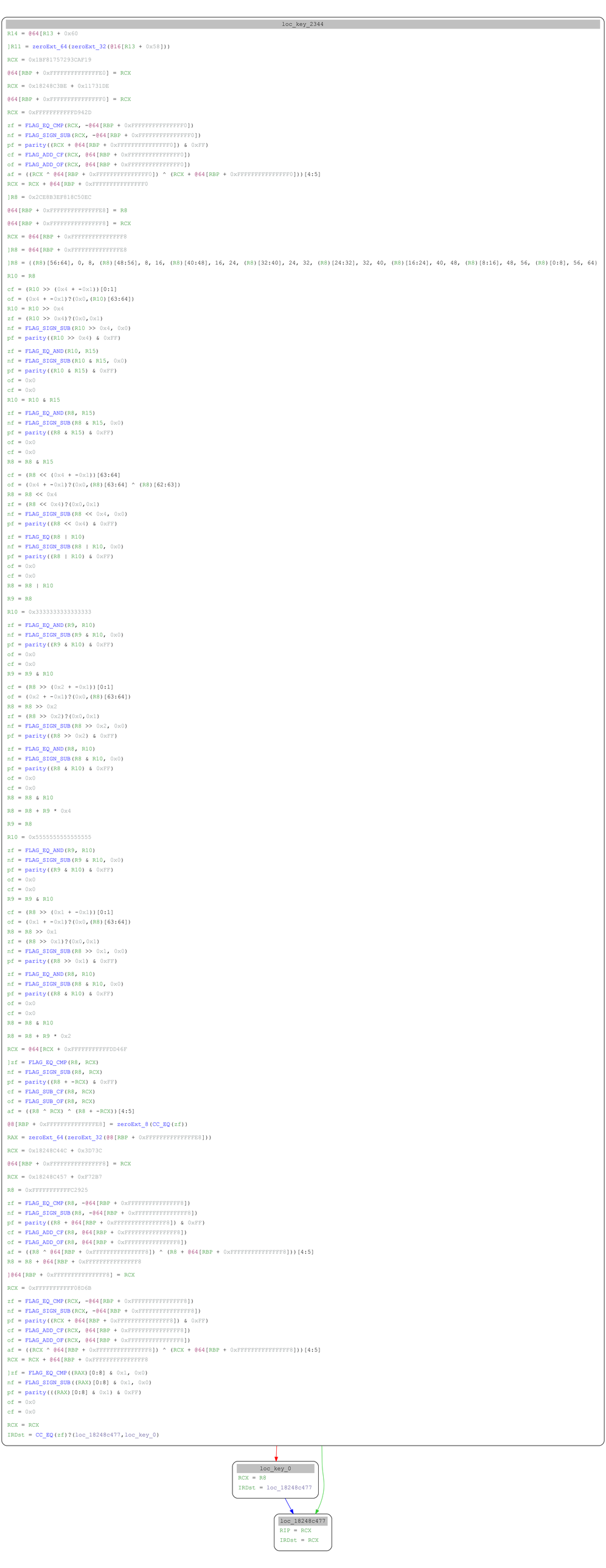
loc_key_2344
R14 = @64[R13 + 0x60]
]R11 = zeroExt_64(zeroExt_32(@16[R13 + 0x58]))
RCX = 0x1BF81757293CAF19
@64[RBP + 0xFFFFFFFFFFFFFFE0] = RCX
RCX = 0x18248C3BE + 0x11731DE
@64[RBP + 0xFFFFFFFFFFFFFFF0] = RCX
RCX = 0xFFFFFFFFFFFD942D
zf = FLAG_EQ_CMP(RCX, -@64[RBP + 0xFFFFFFFFFFFFFFF0])
nf = FLAG_SIGN_SUB(RCX, -@64[RBP + 0xFFFFFFFFFFFFFFF0])
pf = parity((RCX + @64[RBP + 0xFFFFFFFFFFFFFFF0]) & 0xFF)
cf = FLAG_ADD_CF(RCX, @64[RBP + 0xFFFFFFFFFFFFFFF0])
of = FLAG_ADD_OF(RCX, @64[RBP + 0xFFFFFFFFFFFFFFF0])
af = ((RCX ^ @64[RBP + 0xFFFFFFFFFFFFFFF0]) ^ (RCX + @64[RBP + 0xFFFFFFFFFFFFFFF0]))[4:5]
RCX = RCX + @64[RBP + 0xFFFFFFFFFFFFFFF0]
]R8 = 0x2CE8B3EF818C50EC
@64[RBP + 0xFFFFFFFFFFFFFFE8] = R8
@64[RBP + 0xFFFFFFFFFFFFFFF8] = RCX
RCX = @64[RBP + 0xFFFFFFFFFFFFFFF8]
]R8 = @64[RBP + 0xFFFFFFFFFFFFFFE8]
]R8 = {(R8)[56:64], 0, 8, (R8)[48:56], 8, 16, (R8)[40:48], 16, 24, (R8)[32:40], 24, 32, (R8)[24:32], 32, 40, (R8)[16:24], 40, 48, (R8)[8:16], 48, 56, (R8)[0:8], 56, 64}
R10 = R8
cf = (R10 >> (0x4 + -0x1))[0:1]
of = (0x4 + -0x1)?(0x0,(R10)[63:64])
R10 = R10 >> 0x4
zf = (R10 >> 0x4)?(0x0,0x1)
nf = FLAG_SIGN_SUB(R10 >> 0x4, 0x0)
pf = parity((R10 >> 0x4) & 0xFF)
zf = FLAG_EQ_AND(R10, R15)
nf = FLAG_SIGN_SUB(R10 & R15, 0x0)
pf = parity((R10 & R15) & 0xFF)
of = 0x0
cf = 0x0
R10 = R10 & R15
zf = FLAG_EQ_AND(R8, R15)
nf = FLAG_SIGN_SUB(R8 & R15, 0x0)
pf = parity((R8 & R15) & 0xFF)
of = 0x0
cf = 0x0
R8 = R8 & R15
cf = (R8 << (0x4 + -0x1))[63:64]
of = (0x4 + -0x1)?(0x0,(R8)[63:64] ^ (R8)[62:63])
R8 = R8 << 0x4
zf = (R8 << 0x4)?(0x0,0x1)
nf = FLAG_SIGN_SUB(R8 << 0x4, 0x0)
pf = parity((R8 << 0x4) & 0xFF)
zf = FLAG_EQ(R8 | R10)
nf = FLAG_SIGN_SUB(R8 | R10, 0x0)
pf = parity((R8 | R10) & 0xFF)
of = 0x0
cf = 0x0
R8 = R8 | R10
R9 = R8
R10 = 0x3333333333333333
zf = FLAG_EQ_AND(R9, R10)
nf = FLAG_SIGN_SUB(R9 & R10, 0x0)
pf = parity((R9 & R10) & 0xFF)
of = 0x0
cf = 0x0
R9 = R9 & R10
cf = (R8 >> (0x2 + -0x1))[0:1]
of = (0x2 + -0x1)?(0x0,(R8)[63:64])
R8 = R8 >> 0x2
zf = (R8 >> 0x2)?(0x0,0x1)
nf = FLAG_SIGN_SUB(R8 >> 0x2, 0x0)
pf = parity((R8 >> 0x2) & 0xFF)
zf = FLAG_EQ_AND(R8, R10)
nf = FLAG_SIGN_SUB(R8 & R10, 0x0)
pf = parity((R8 & R10) & 0xFF)
of = 0x0
cf = 0x0
R8 = R8 & R10
R8 = R8 + R9 * 0x4
R9 = R8
R10 = 0x5555555555555555
zf = FLAG_EQ_AND(R9, R10)
nf = FLAG_SIGN_SUB(R9 & R10, 0x0)
pf = parity((R9 & R10) & 0xFF)
of = 0x0
cf = 0x0
R9 = R9 & R10
cf = (R8 >> (0x1 + -0x1))[0:1]
of = (0x1 + -0x1)?(0x0,(R8)[63:64])
R8 = R8 >> 0x1
zf = (R8 >> 0x1)?(0x0,0x1)
nf = FLAG_SIGN_SUB(R8 >> 0x1, 0x0)
pf = parity((R8 >> 0x1) & 0xFF)
zf = FLAG_EQ_AND(R8, R10)
nf = FLAG_SIGN_SUB(R8 & R10, 0x0)
pf = parity((R8 & R10) & 0xFF)
of = 0x0
cf = 0x0
R8 = R8 & R10
R8 = R8 + R9 * 0x2
RCX = @64[RCX + 0xFFFFFFFFFFFDD46F]
]zf = FLAG_EQ_CMP(R8, RCX)
nf = FLAG_SIGN_SUB(R8, RCX)
pf = parity((R8 + -RCX) & 0xFF)
cf = FLAG_SUB_CF(R8, RCX)
of = FLAG_SUB_OF(R8, RCX)
af = ((R8 ^ RCX) ^ (R8 + -RCX))[4:5]
@8[RBP + 0xFFFFFFFFFFFFFFE8] = zeroExt_8(CC_EQ(zf))
RAX = zeroExt_64(zeroExt_32(@8[RBP + 0xFFFFFFFFFFFFFFE8]))
RCX = 0x18248C44C + 0x3D73C
@64[RBP + 0xFFFFFFFFFFFFFFF8] = RCX
RCX = 0x18248C457 + 0xF72B7
R8 = 0xFFFFFFFFFFFC2925
zf = FLAG_EQ_CMP(R8, -@64[RBP + 0xFFFFFFFFFFFFFFF8])
nf = FLAG_SIGN_SUB(R8, -@64[RBP + 0xFFFFFFFFFFFFFFF8])
pf = parity((R8 + @64[RBP + 0xFFFFFFFFFFFFFFF8]) & 0xFF)
cf = FLAG_ADD_CF(R8, @64[RBP + 0xFFFFFFFFFFFFFFF8])
of = FLAG_ADD_OF(R8, @64[RBP + 0xFFFFFFFFFFFFFFF8])
af = ((R8 ^ @64[RBP + 0xFFFFFFFFFFFFFFF8]) ^ (R8 + @64[RBP + 0xFFFFFFFFFFFFFFF8]))[4:5]
R8 = R8 + @64[RBP + 0xFFFFFFFFFFFFFFF8
]@64[RBP + 0xFFFFFFFFFFFFFFF8] = RCX
RCX = 0xFFFFFFFFFF08D6B
zf = FLAG_EQ_CMP(RCX, -@64[RBP + 0xFFFFFFFFFFFFFFF8])
nf = FLAG_SIGN_SUB(RCX, -@64[RBP + 0xFFFFFFFFFFFFFFF8])
pf = parity((RCX + @64[RBP + 0xFFFFFFFFFFFFFFF8]) & 0xFF)
cf = FLAG_ADD_CF(RCX, @64[RBP + 0xFFFFFFFFFFFFFFF8])
of = FLAG_ADD_OF(RCX, @64[RBP + 0xFFFFFFFFFFFFFFF8])
af = ((RCX ^ @64[RBP + 0xFFFFFFFFFFFFFFF8]) ^ (RCX + @64[RBP + 0xFFFFFFFFFFFFFFF8]))[4:5]
RCX = RCX + @64[RBP + 0xFFFFFFFFFFFFFFF8
]zf = FLAG_EQ_CMP((RAX)[0:8] & 0x1, 0x0)
nf = FLAG_SIGN_SUB((RAX)[0:8] & 0x1, 0x0)
pf = parity(((RAX)[0:8] & 0x1) & 0xFF)
of = 0x0
cf = 0x0
RCX = RCX
IRDst = CC_EQ(zf)?(loc_18248c477,loc_key_0)
loc_key_0
RCX = R8
IRDst = loc_18248c477
loc_18248c477
RIP = RCX
IRDst = RCX