BCH重启OpCodes背后的安全故事

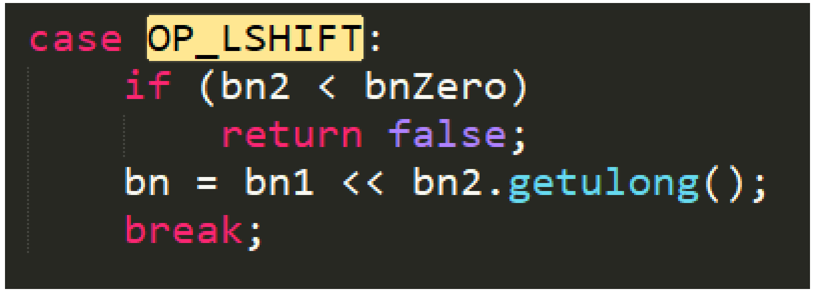
彭峙酿
246003@qq.com

关于我

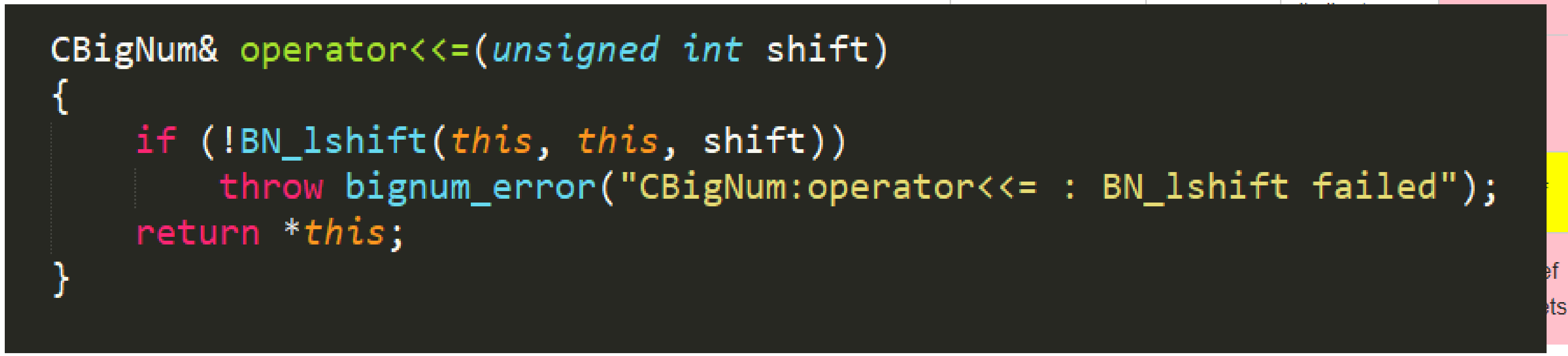
- PhD. 密码学
- 360 核心安全
- 关注领域：
    - 数据安全
    - 区块链安全
    - 软件安全

OP_LSHIFT crash

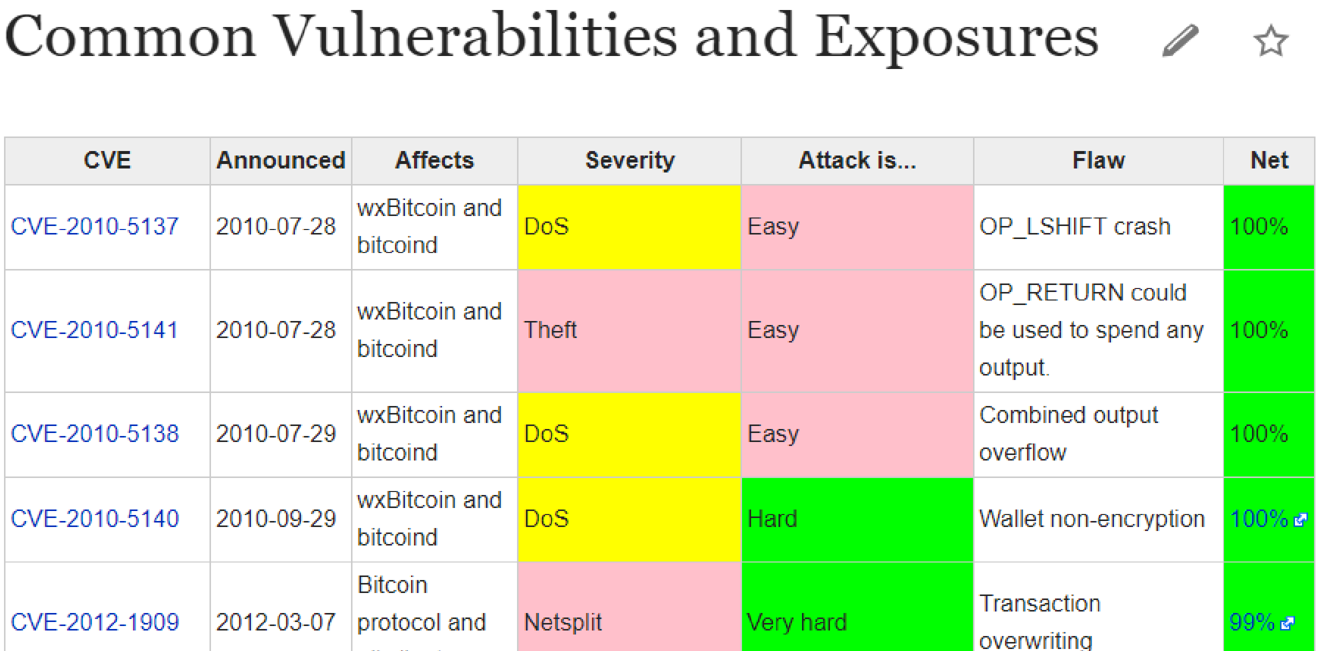
- 有记录的第一个bitcoin漏洞
- 整个bitcoin测试网络崩溃



```
case OP_LSHIFT:
    if (bn2 < bnZero)
        return false;
    bn = bn1 << bn2.getulong();
    break;
```



```
CBigNum& operator<<=(unsigned int shift)
{
    if (!BN_lshift(this, this, shift))
        throw bignum_error("CBigNum:operator<<= : BN_lshift failed");
    return *this;
}
```

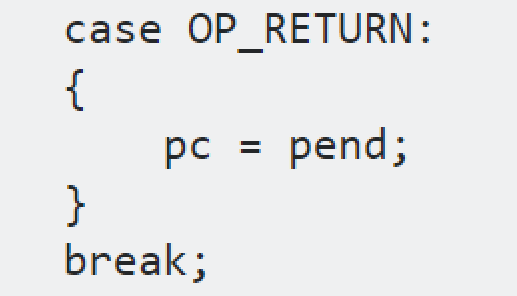


Common Vulnerabilities and Exposures

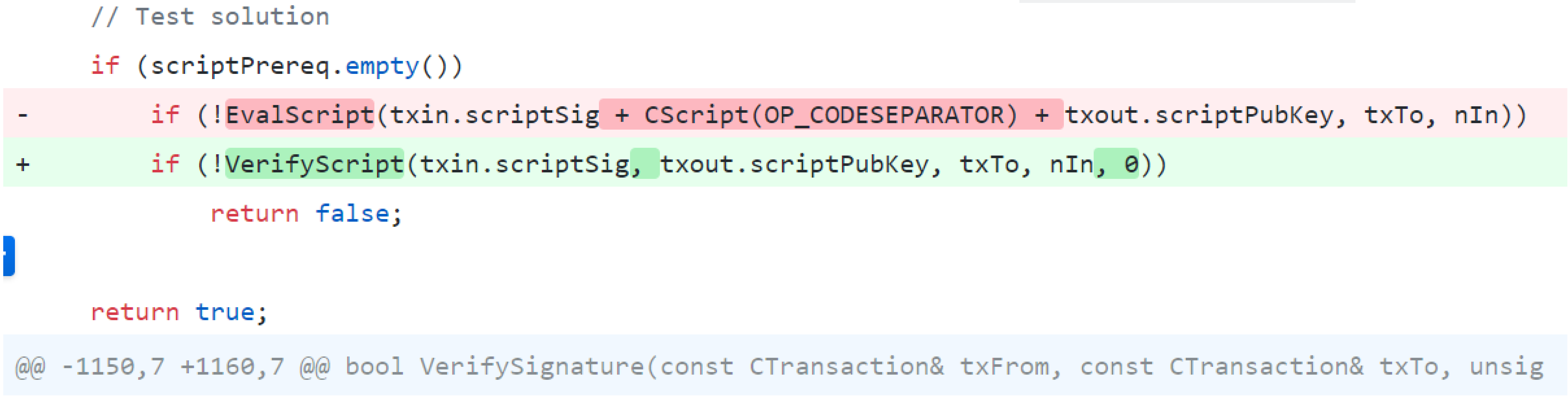
CVE	Announced	Affects	Severity	Attack is...	Flaw	Net
CVE-2010-5137	2010-07-28	wxBitcoin and bitcoind	DoS	Easy	OP_LSHIFT crash	100%
CVE-2010-5141	2010-07-28	wxBitcoin and bitcoind	Theft	Easy	OP_RETURN could be used to spend any output.	100%
CVE-2010-5138	2010-07-29	wxBitcoin and bitcoind	DoS	Easy	Combined output overflow	100%
CVE-2010-5140	2010-09-29	wxBitcoin and bitcoind	DoS	Hard	Wallet non-encryption	100%
CVE-2012-1909	2012-03-07	Bitcoin protocol and	Netsplit	Very hard	Transaction overwriting	99%
				Hard	MingW non-multithreading	100%
				Miners	Mandatory P2SH protocol update	99%
			ef tsplit"/>	Easy	Block hash collision (via merkle root)	99% <!--

OP_Return

- The 1 Return BUG
- 任意UTXO花费：OP_1 OP_RETURN



```
case OP_RETURN:
{
    pc = pend;
}
break;
```



```
    // Test solution
    if (scriptPrereq.empty())
-       if (!EvalScript(txin.scriptSig + CScript(OP_CODESEPARATOR) + txout.scriptPubKey, txTo, nIn))
+       if (!VerifyScript(txin.scriptSig, txout.scriptPubKey, txTo, nIn, 0))
            return false;

    return true;
@@ -1150,7 +1160,7 @@ bool VerifySignature(const CTransaction& txFrom, const CTransaction& txTo, unsig
```

OP_CAT OP_DUP 组合攻击

- OP_CAT OP_DUP组合使用
- 栈大小double
- 10次调用32KB
- 20次调用32MB
- …
- 40次调用32TB

- Stack: A (length=32)

  OP_DUP

  Stack: A A

  OP_CAT

  Stack: AA (length=64)

  OP_DUP

  Stack: AA AA

  OP_CAT

  Stack: AAAA (length=128)

  OP_DUP

  Stack: AAAA AAAA

  OP_CAT

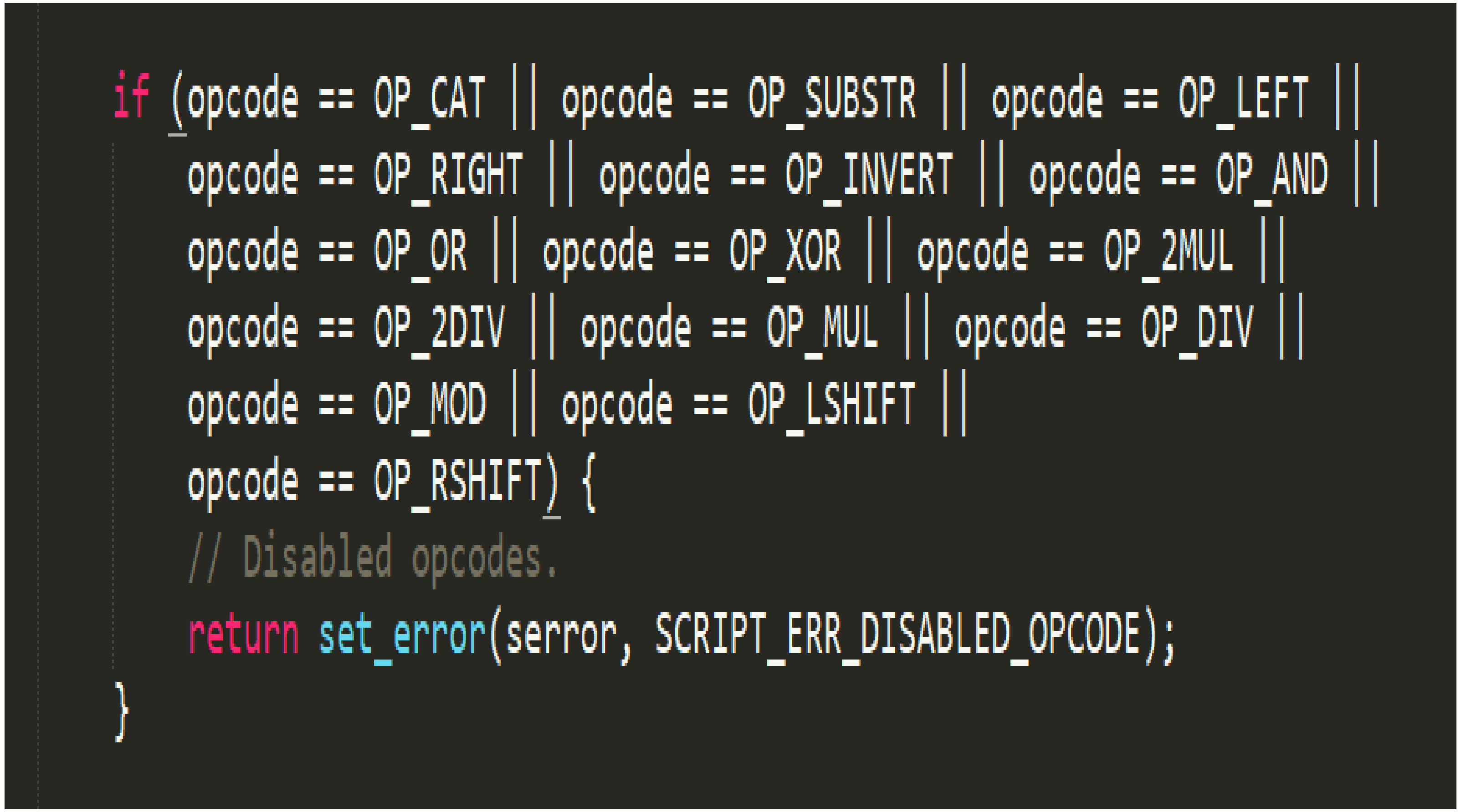
  Stack: AAAAAAAA (length=256)

多个脚本被禁用----解禁

- 功能性 vs 安全性

解决办法：

安全设计
模糊测试fuzz
代码审计
测试网络

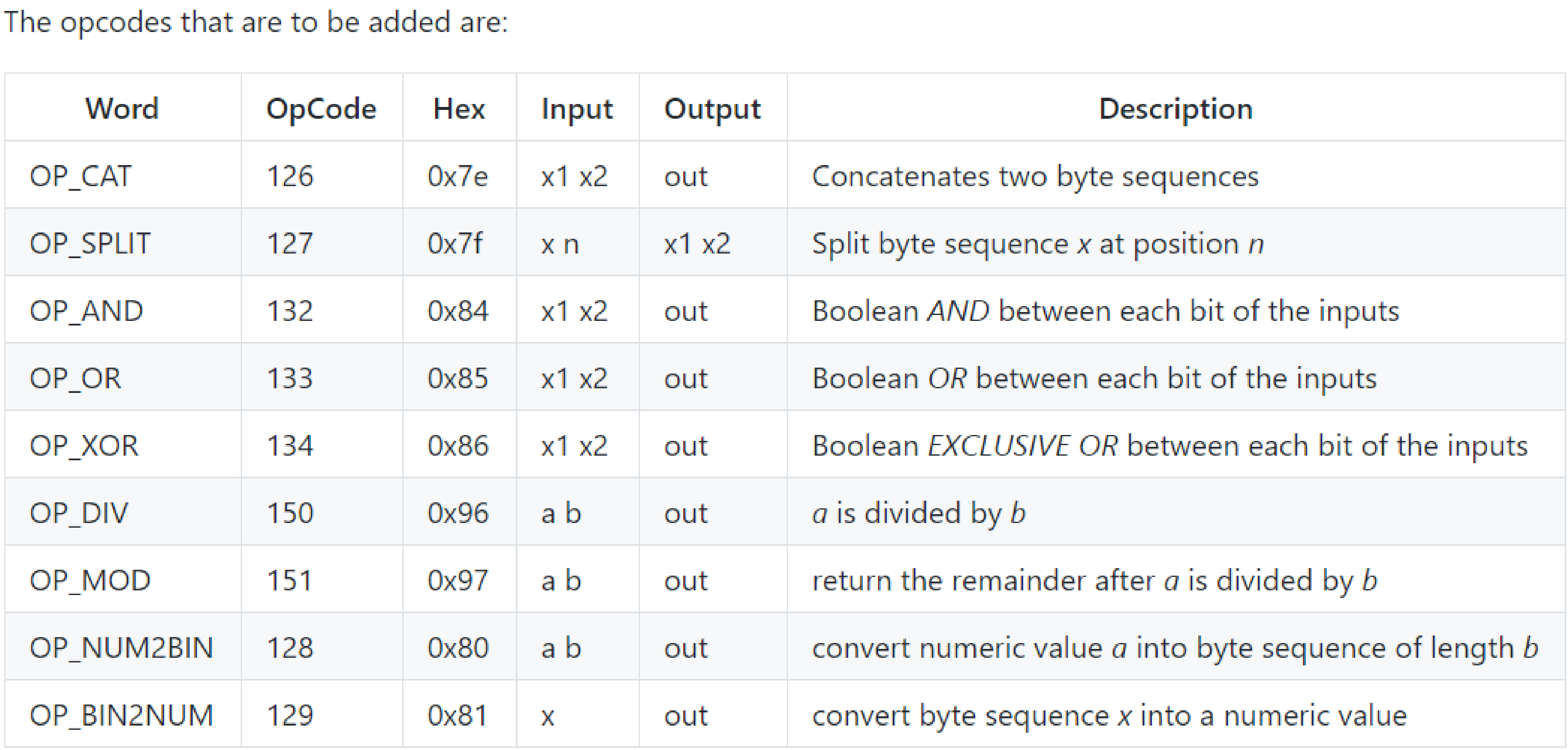


```
if (opcode == OP_CAT || opcode == OP_SUBSTR || opcode == OP_LEFT ||
    opcode == OP_RIGHT || opcode == OP_INVERT || opcode == OP_AND ||
    opcode == OP_OR || opcode == OP_XOR || opcode == OP_2MUL ||
    opcode == OP_2DIV || opcode == OP_MUL || opcode == OP_DIV ||
    opcode == OP_MOD || opcode == OP_LSHIFT ||
    opcode == OP_RSHIFT) {
    // Disabled opcodes.
    return set_error(serror, SCRIPT_ERR_DISABLED_OPCODE);
}
```

区块链代码审计

- Javascript解析器漏洞
  - 脚本功能强
  - 堆风水布局
  - JIT过度优化
- 智能合约解析器
  - 功能相对简单
  - JIT优化多数未开启
  - 恶魔在细节中
- 360核心安全区块链相关
  - 矿池　影响Zcash BTG Hush等 CVE-2018-10831
  - 全节点 BTC,以太坊,门罗币等
  - 钱包　多币种 CVE-2018-10812
- BitcoinCandy (CDY)安全经验
  - 机枪池
  - Fake Miners

BCH重启部分OpCodes



The opcodes that are to be added are:

Word	OpCode	Hex	Input	Output	Description
OP_CAT	126	0x7e	x1 x2	out	Concatenates two byte sequences
OP_SPLIT	127	0x7f	x n	x1 x2	Split byte sequence *x* at position *n*
OP_AND	132	0x84	x1 x2	out	Boolean *AND* between each bit of the inputs
OP_OR	133	0x85	x1 x2	out	Boolean *OR* between each bit of the inputs
OP_XOR	134	0x86	x1 x2	out	Boolean *EXCLUSIVE OR* between each bit of the inputs
OP_DIV	150	0x96	a b	out	*a* is divided by *b*
OP_MOD	151	0x97	a b	out	return the remainder after *a* is divided by *b*
OP_NUM2BIN	128	0x80	a b	out	convert numeric value *a* into byte sequence of length *b*
OP_BIN2NUM	129	0x81	x	out	convert byte sequence *x* into a numeric value

谢谢！