Big numbers in OpenSSL

CATALDO BASILE

< CATALDO.BASILE@ POLITO.IT >

POLITECNICO DI TORINO

Agenda

- topics
  - OpenSSL BIGNUM
    - represent large arbitrary precision integers
    - including RSA and DH data structures
- ...and some examples in C

BIGNUM – Arbitrary Precision Math

- public-key cryptography handles very large integers (>1024 bit)
  - standard C data types are not large enough (32/64 bit)
- BIGNUM package supports integers of any size (no upper bounds)
  - #include <openssl/bn.h>
- BIGNUM = object/data structure (or context)
  - memory allocated dynamically to cope with the representation needs
  - https://www.openssl.org/docs/manmaster/man3/BN_new.html

```
BIGNUM *bn;

/* allocate a dynamic BIGNUM */
bn = BN_new();

/* free the BIGNUMs */
BN_free(dynamic_bn);
```

RSA keys: data structure using BIGNUMs

- asymmetric keys represented via a data structure

```
struct {
            BIGNUM *n; // public modulus
            BIGNUM *e; // public exponent
            BIGNUM *d; // private exponent
            BIGNUM *p; // secret prime factor
            BIGNUM *q; // secret prime factor
            BIGNUM *dmp1; // d mod (p-1)
            BIGNUM *dmq1; // d mod (q-1)
            BIGNUM *iqmp; // q^-1 mod p
}; RSA
```

Copying BIGNUMs

- deep copy is required when with BIGNUMs
  - ...a typical issue with pointers in C

```
BIGNUM *a, *b, *c;

/* wrong way */
a = b;

/* right way to copy a BIGNUM */
BN_copy(a, b); /* copies b to a */
c = BN_dup(b); /* creates c and initializes it to the same value
as b */
```

BIGNUM conversions

- convert a BIGNUM into its binary representation
  - to store (e.g., save to a file)
  - to send (e.g., via a socket)
- convert a BIGNUM to a decimal or hexadecimal representation
  - to print it
- https://www.openssl.org/docs/manmaster/man3/BN_bn2bin.html

```
BIGNUM *num;

/* converting from BIGNUM to binary */
len = BN_num_bytes(num);
buf = (unsigned char*)calloc(len,sizeof(unsigned char));
len = BN_bn2bin(num, buf);

/* converting from binary to BIGNUM */
BN_bin2bn(buf, len, num);
num = BN_bin2bn(buf, len, NULL);
```

BIGNUM: operations

arithmetic operations

- int BN_add(BIGNUM *r, const BIGNUM *a, const BIGNUM *b);
- int BN_sqr(BIGNUM *r, BIGNUM *a, BN_CTX *ctx);
- int BN_div(BIGNUM *dv, BIGNUM *rem, const BIGNUM *a, const BIGNUM *d, BN_CTX *ctx);
- int BN_mod_add(BIGNUM *r, BIGNUM *a, BIGNUM *b, const BIGNUM *m, BN_CTX *ctx);
- int BN_mod_exp(BIGNUM *r, BIGNUM *a, const BIGNUM *p, const BIGNUM *m, BN_CTX *ctx);
- int BN_gcd(BIGNUM *r, BIGNUM *a, BIGNUM *b, BN_CTX *ctx);
- https://www.openssl.org/docs/manmaster/man3/BN_add.html

tests / logical operations

- int BN_cmp(const BIGNUM *a, const BIGNUM *b);
- int BN_is_zero(const BIGNUM *a); int BN_is_one(const BIGNUM *a);
- int BN_is_word(const BIGNUM *a, const BN_ULONG w);
- https://www.openssl.org/docs/manmaster/man3/BN_cmp.html

Additional resources

random

- https://www.openssl.org/docs/man1.0.2/man3/BN_rand.html

operations

- https://www.openssl.org/docs/man1.0.2/man3/BN_add.html
- https://www.openssl.org/docs/man1.0.2/man3/BN_lshift.html

comparison

- https://www.openssl.org/docs/man1.0.2/man3/BN_cmp.html

primes

- https://www.openssl.org/docs/man1.0.2/man3/BN_generate_prime.html