${}_{C}TX*$
$ctx =$
$BN_{C}TX_{n}ew()$
${}_{n}ew()$
${}_{d}ec2bn(a, 12345678901112231223”); //AssignavaluefromahexnumberstringBN_{h}ex2bn(a, 2A3B4C55FF77889AED3F”$
${}_{s}tr =$
$BN_{b}n2dec(a); //Printoutthenumberstringprintf(”//FreethedynamicallyallocatedmemoryOPENSSL_{f}ree(number_{s}tr$
$res =$
$abres =$
$a+$
$b :$
${}_{s}ub(res, a, b); BN_{a}dd(res, a, b);$
$res =$
$a*$
$b.$
${}_{m}ul(res, a, b, ctx)$
$res =$
$a*$
$b*$
$mod(n) :$
${}_{m}od_{m}ul(res, a, b, n, ctx)$
$res =$
$a^{c}*$
$mod(n) :$
${}_{m}od_{e}xp(res, a, c, n, ctx)$
$a*$
$b*$
$mod(n) =$
$1(a*$
$b1*$
$mod(n))$
${}_{m}od_{i}nverse(b, a, n, ctx);$
$a*$
b
$(a*$
$b*$
$mod(n)$
${}_{s}ample.c*$
$/include <$
$stdio.h >$
$include <$
$openssl/bn.h >$
$defineNBITS256voidprintBN(char*$
$msg, BIGNUM*$
$a)/*UseBN_{b}n2hex(a)forhexstring*UseBN_{b}n2dec(a)fordecimalstring*/char*number_{s}tr = BN_{b}n2hex(a); printf($
${}_{C}TX*$
$ctx =$
$BN_{C}TX_{n}ew();$
${}_{n}ew(); BIGNUM*$
$b =$
$BN_{n}ew(); BIGNUM*$
$n =$
$BN_{n}ew(); BIGNUM*$
$res =$
$BN_{n}ew();$
${}_{g}enerate_{p}rime_{e}x(a, NBITS, 1, NULL, NULL, NULL); BN_{d}ec2bn(b, ”273489463796838501848592769467194369268”); B$
${}_{m}ul(res, a, b, ctx); printBN(”a*$
$b =$
$”, res);$
${}^{b}modnBN_{m}od_{e}xp(res, a, b, n, ctx); printBN(”acmodn =$
$”, res); return0;$
$gccbn_{s}ample.c-$
$lcrypto$
$e*$
$d \equiv$
$1*$
$mod((p-$
$1)*$
$(q-$
$1)) \Rightarrow$
$e*$
$d*$
$mod((p-$
$1)*$
$(q-$
$1)) =$
1
BIGNUM
APIs
BIGNUM
BIGNUM
$python-$
$c'print(”Atopsecret!”.encode(”hex”))'$
$C =$
$M^{e}mod(n)$
$M =$
$C^{d}mod(n)$
BN_CTX
BN_CTX
$python-$
$c'print(”4120746f702073656372657421”.decode(”hex”))'$
2000

$/C =$
$US/O =$
$DigiCertInc/OU =$
$www.digicert.com/CN =$
$DigiCertSHA2HighAssuranceServerCA-$
$--$
$--$
$BEGINCERTIFICATE-$
$--$
$--$
$MIIF8jCCBNqgAwIBAgIQDmTF+$
$8I2reFLFyrrQceMsDANBgkqhkiG9w0BAQsFADBwMQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGl$
$--$
$--$
$-ENDCERTIFICATE-$
$--$
$--$

B μ
2:
E
μ
(e,
n)