**SUSPICIOUS COMMENT: Comment Indicates Potentially Unfinished Code -**

Line: 3 - C:\Users\Jeremy\Downloads\x86\_64-gcc.c

kind of dirty hack for Sun Studio \*/

**MEDIUM: Potentially Unsafe Code - goto**

Line: 185 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_add(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 189 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_sub(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 193 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_lshift1(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 198 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 202 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_lshift(out,ctx,NULL)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 206 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_rshift1(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 210 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_rshift(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 214 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_sqr(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 218 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_mul(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 222 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_div(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 226 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_div\_word(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 230 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_div\_recp(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 234 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_mod(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 238 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_mod\_mul(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 242 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_mont(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 246 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_mod\_exp(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 250 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_mod\_exp\_mont\_consttime(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 251 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_mod\_exp\_mont5(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 255 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_exp(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 259 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_kron(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 263 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_sqrt(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 267 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_small\_prime(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 274 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_probable\_prime\_coprime(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 280 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_add(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 284 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod(out)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 288 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod\_mul(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 292 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod\_sqr(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 296 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod\_inv(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 300 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod\_div(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 304 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod\_exp(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 308 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod\_sqrt(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 312 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!test\_gf2m\_mod\_solve\_quad(out,ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 701 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 726 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 751 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 775 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1240 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1247 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1301 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1369 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1432 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1487 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1548 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1619 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1678 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1742 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1768 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1806 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (a == NULL || b == NULL || r == NULL || t == NULL) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1819 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_generate\_prime\_ex(b, 512, 0, NULL, NULL, &cb)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1825 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_bntest\_rand(a, 512, 0, 0)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1829 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_copy(t, b)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1831 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_sub\_word(t, 1)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1832 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_rshift1(t, t)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1836 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_mod\_exp\_recp(r, a, t, b, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1845 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_add\_word(r, 1)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1849 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1855 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (kronecker < -1) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1867 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1895 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (a == NULL || p == NULL || r == NULL) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1905 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_set\_word(p, primes[i])) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1909 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_set\_word(a, 32)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1910 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_set\_word(r, 2\*i + 1)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1912 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_generate\_prime\_ex(p, 256, 0, a, r, &cb)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1921 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_bntest\_rand(r, 256, 0, 3)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1922 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_nnmod(r, r, p, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1923 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_mod\_sqr(r, r, p, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1924 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_bntest\_rand(a, 256, 0, 3)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1925 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_nnmod(a, a, p, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1926 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_mod\_sqr(a, a, p, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1927 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_mul(a, a, r, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1929 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_sub(a, a, p)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1931 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_mod\_sqrt(r, a, p, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1932 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_mod\_sqr(r, r, p, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1934 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!BN\_nnmod(a, a, p, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1945 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1971 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1975 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 1995 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

if (!bn\_probable\_prime\_dh\_coprime(&r, 1024, ctx)) goto err;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 2004 - C:\Users\Jeremy\Downloads\bntest.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto err;

**STANDARD: Potential Memory Mis-management. Variable Name: &a**

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1252 FileName: C:\Users\Jeremy\Downloads\bntest.c

**STANDARD: Potential Memory Mis-management. Variable Name: &b**

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1252 FileName: C:\Users\Jeremy\Downloads\bntest.c

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1253 FileName: C:\Users\Jeremy\Downloads\bntest.c

**STANDARD: Potential Memory Mis-management. Variable Name: &c**

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1252 FileName: C:\Users\Jeremy\Downloads\bntest.c

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1253 FileName: C:\Users\Jeremy\Downloads\bntest.c

7 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1254 FileName: C:\Users\Jeremy\Downloads\bntest.c

**STANDARD: Potential Memory Mis-management. Variable Name: &d**

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1252 FileName: C:\Users\Jeremy\Downloads\bntest.c

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1253 FileName: C:\Users\Jeremy\Downloads\bntest.c

7 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1254 FileName: C:\Users\Jeremy\Downloads\bntest.c

4 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 856 FileName: C:\Users\Jeremy\Downloads\bntest.c

**STANDARD: Potential Memory Mis-management. Variable Name: &e**

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1252 FileName: C:\Users\Jeremy\Downloads\bntest.c

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1253 FileName: C:\Users\Jeremy\Downloads\bntest.c

7 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1254 FileName: C:\Users\Jeremy\Downloads\bntest.c

4 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 856 FileName: C:\Users\Jeremy\Downloads\bntest.c

3 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 685 FileName: C:\Users\Jeremy\Downloads\bntest.c

**STANDARD: Potential Memory Mis-management. Variable Name: mont**

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1252 FileName: C:\Users\Jeremy\Downloads\bntest.c

8 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1253 FileName: C:\Users\Jeremy\Downloads\bntest.c

7 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 1254 FileName: C:\Users\Jeremy\Downloads\bntest.c

4 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 856 FileName: C:\Users\Jeremy\Downloads\bntest.c

3 free

Multiple frees detected. Check code paths manually to ensure that variables cannot be freed more than once.

Line: 685 FileName: C:\Users\Jeremy\Downloads\bntest.c

new without delete.

Line: 1097 FileName: C:\Users\Jeremy\Downloads\bntest.c

**MEDIUM: Potentially Unsafe Code - assert**

Line: 64 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

#include <assert.h>

**MEDIUM: Potentially Unsafe Code - assert**

Line: 75 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(num >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 101 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(num >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 124 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(n >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 151 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(num >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 180 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(num >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 206 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(n >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 247 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert((i == BN\_BITS2) || (h <= (BN\_ULONG)1<<i));

**MEDIUM: Potentially Unsafe Code - assert**

Line: 310 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(n >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 345 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(n >= 0);

**MEDIUM: Potentially Unsafe Code - assert**

Line: 398 - C:\Users\Jeremy\Downloads\bn\_asm.c

The 'assert' macro usually only exists for code in the debug build. In general, no check will take place in production code. Verify that this check does not perform any critical function and is not being used in place of error handling.

assert(n >= 0);

**STANDARD: Potentially Unsafe Code - alloca**

Line: 836 - C:\Users\Jeremy\Downloads\bn\_asm.c

Function appears in Microsoft's banned function list. Can facilitate buffer overflow conditions and other memory mis-management situations.

#include <alloca.h>

**STANDARD: Potentially Unsafe Code - alloca**

Line: 862 - C:\Users\Jeremy\Downloads\bn\_asm.c

Function appears in Microsoft's banned function list. Can facilitate buffer overflow conditions and other memory mis-management situations.

vp = tp = alloca((num+2)\*sizeof(BN\_ULONG));

**MEDIUM: Potentially Unsafe Code - goto**

Line: 880 - C:\Users\Jeremy\Downloads\bn\_asm.c

Use of 'goto' function. The goto function can result in unstructured code which is difficult to maintain and can result in failures to initialise or de-allocate memory.

goto enter;

**SUSPICIOUS COMMENT: Comment Indicates Potentially Unfinished Code -**

Line: 951 - C:\Users\Jeremy\Downloads\bn\_asm.c

hmm... is it faster just to do a multiply? \*/

**STANDARD: Potentially Unsafe Code - alloca**

Line: 988 - C:\Users\Jeremy\Downloads\bn\_asm.c

Function appears in Microsoft's banned function list. Can facilitate buffer overflow conditions and other memory mis-management situations.

#include <alloca.h>

**STANDARD: Potentially Unsafe Code - alloca**

Line: 995 - C:\Users\Jeremy\Downloads\bn\_asm.c

Function appears in Microsoft's banned function list. Can facilitate buffer overflow conditions and other memory mis-management situations.

vp = tp = alloca((num+2)\*sizeof(BN\_ULONG));