**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 156 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(&replay, d, sizeof(replay));

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 205 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(&secretid, d, sizeof(secretid));

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 221 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(&secretid, d, sizeof(secretid));

**MEDIUM: Potentially Unsafe Code - malloc( ) Using Fixed Value Instead of Variable Type Size**

Line: 243 - C:\Users\Jeremy\Downloads\auth(1).c

The code uses a fixed value for malloc instead of the variable type size which is dependent on the platform (e.g. sizeof(int) instead of '4'). This can result in too much or too little memory being assigned with unpredicatble results such as performance impact, overflows or memory corruption.

state->reconf->key = malloc(16);

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 255 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(state->reconf->key, d, 16);

**MEDIUM: Potentially Unsafe Code - goto**

Line: 281 - C:\Users\Jeremy\Downloads\auth(1).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 gottoken;

**MEDIUM: Potentially Unsafe Code - goto**

Line: 329 - C:\Users\Jeremy\Downloads\auth(1).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 finish;

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 336 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(mm, m, mlen);

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 378 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(state->token->key, t->key, t->key\_len);

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 388 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(state->token->realm, t->realm,

**STANDARD: Potentially Unsafe Code - fopen**

Line: 418 - C:\Users\Jeremy\Downloads\auth(1).c

Function used to open file. Carry out a manual check to ensure that user cannot modify filename for malicious purposes and that file is not 'opened' more than once simultaneously.

fp = fopen(RDM\_MONOFILE, "r+");

**STANDARD: Potentially Unsafe Code - fopen**

Line: 422 - C:\Users\Jeremy\Downloads\auth(1).c

Function used to open file. Carry out a manual check to ensure that user cannot modify filename for malicious purposes and that file is not 'opened' more than once simultaneously.

fp = fopen(RDM\_MONOFILE, "w");

**HIGH: Potentially Unsafe Code - fscanf**

Line: 433 - C:\Users\Jeremy\Downloads\auth(1).c

Function appears in Microsoft's banned function list. The function directs external input to a buffer and so can facilitate buffer overflows.

if (fscanf(fp, "0x%016" PRIu64, &rdm) != 1)

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

Line: 564 - C:\Users\Jeremy\Downloads\auth(1).c

\* We only need to do this for DISCOVER messages

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 620 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(data, &rdm, 8);

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 639 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(data, t->key, t->key\_len);

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 657 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(data, t->realm, t->realm\_len);

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 671 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(data, &secretid, sizeof(secretid));

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 685 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(&giaddr, p, sizeof(giaddr));

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 698 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(data, hmac\_code, sizeof(hmac\_code));

**MEDIUM: Potentially Unsafe Code - memcpy**

Line: 707 - C:\Users\Jeremy\Downloads\auth(1).c

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

memcpy(p, &giaddr, sizeof(giaddr));

**STANDARD: Potential Memory Mis-management. Variable Name: state->token->realm**

malloc without free.

Line: 385 FileName: C:\Users\Jeremy\Downloads\auth(1).c

**STANDARD: Potential Memory Mis-management. Variable Name: state->token**

malloc without free.

Line: 385 FileName: C:\Users\Jeremy\Downloads\auth(1).c

2 free

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

Line: 392 FileName: C:\Users\Jeremy\Downloads\auth(1).c

**STANDARD: Potential Memory Mis-management. Variable Name: state->reconf->key**

malloc without free.

Line: 385 FileName: C:\Users\Jeremy\Downloads\auth(1).c

2 free

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

Line: 392 FileName: C:\Users\Jeremy\Downloads\auth(1).c

malloc without free.

Line: 243 FileName: C:\Users\Jeremy\Downloads\auth(1).c

**STANDARD: Potential Memory Mis-management. Variable Name: state->reconf**

malloc without free.

Line: 385 FileName: C:\Users\Jeremy\Downloads\auth(1).c

2 free

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

Line: 392 FileName: C:\Users\Jeremy\Downloads\auth(1).c

malloc without free.

Line: 243 FileName: C:\Users\Jeremy\Downloads\auth(1).c

2 free

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

Line: 245 FileName: C:\Users\Jeremy\Downloads\auth(1).c

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

malloc without free.

Line: 385 FileName: C:\Users\Jeremy\Downloads\auth(1).c

2 free

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

Line: 392 FileName: C:\Users\Jeremy\Downloads\auth(1).c

malloc without free.

Line: 243 FileName: C:\Users\Jeremy\Downloads\auth(1).c

2 free

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

Line: 245 FileName: C:\Users\Jeremy\Downloads\auth(1).c

2 free

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

Line: 360 FileName: C:\Users\Jeremy\Downloads\auth(1).c