"scanf()" and "fscanf()" format strings should specify a field width for the "%s" string placeholder

Vulnerability
Critical

- cwe
- sans-top25
- owasp
- injection

The `%s` placeholder is used to read a word into a string.

By default, there is no restriction on the length of that word, and the developer is required to pass a sufficiently large buffer for storing it.

No matter how large the buffer is, there will always be a longer word.

Therefore, programs relying on `%s` are vulnerable to buffer overflows.

A field width specifier can be used together with the `%s` placeholder to limit the number of bytes which will by written to the buffer.

Note that an additional byte is required to store the null terminator.

Noncompliant Code Example

```
char buffer[10];
scanf("%s", buffer);      // Noncompliant - will overflow when a word
longer than 9 characters is entered
```

Compliant Solution

```
char buffer[10];
scanf("%9s", buffer);     // Compliant - will not overflow
```

See

- OWASP Top 10 2017 Category A9 - Using Components with Known Vulnerabilities
- MITRE, CWE-120 - Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
- MITRE, CWE-676 - Use of Potentially Dangerous Function
- SANS Top 25 - Risky Resource Management