Rapid window title changes cause 'white screen of death'

By
Bill Toulas (https://www.bleepingcomputer.com/author/bill-toulas/)

January 8, 2022 10:16 AM **0**

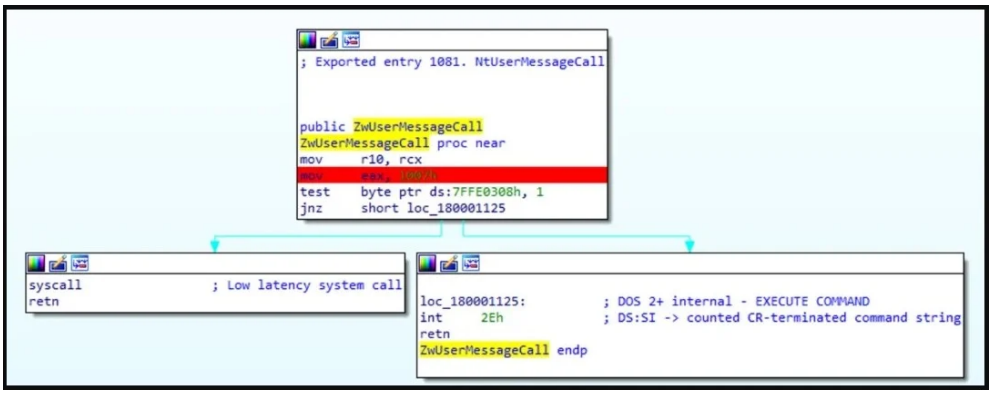
Experimentation with ANSI escape characters on terminal emulators has led to the discovery of multiple high-severity DoS (denial of service) vulnerabilities on Windows terminals and Chrome-based web browsers.

Eviatar Gerzi, a security researcher at CyberArk, has tried out various potential abuse pathways based on an old 2003 advisory (https://marc.info/?l=bugtraq&m=104612710031920&q=p3) on code

execution via window title modifications and discovered a way to induce rapid window title changes on PuTTY.

This atypical attack caused the test machine to enter a state known as the "White Screen of Death", where everything freezes except for the mouse cursor.

Upon testing a similar attack on a local application, the system entered WSOD immediately due to overburdening the OS kernel with calls.



Calls overwhelming the system kernel

Source: CyberArk

The abused function is 'SetWindowText (https://docs.microsoft.com/en-us/windows/win32/api/winuser/nf-winuser-setwindowtexta),' which allows changing the text of the specified window's title bar.

The only way out of the WSOD state is to restart the computer, so this simple trick can lead to a DoS state on a range of applications.

```
static void wintw_set_title(TermWin *tw, const char *title)
{
    sfree(window_name);
    window_name = snewn(1 + strlen(title), char);
    strcpy(window_name, title);
    if (conf_get_bool(conf, CONF_win_name_always) || !IsIconic(hwnd))
        SetWindowText(hwnd, title);
}
```

SetWindowText function in PuTTY

Source: CyberArk

As the researcher points out (https://www.cyberark.com/resources/threat-research-blog/dont-trust-this-title-abusing-terminal-emulators-with-ansi-escape-characters), 'SetWindowText' isn't the only possible leverage for hung ups, as discovered in the case of MobaXterm.

> *In one of the cases, I tested the MobaXterm terminal, and I was surprised that it didn't use SetWindowText function to change the window title but, rather, a function named GdipDrawString (https://docs.microsoft.com/en-us/previous-versions/ms535991(v=vs.85)).*
>
> *The interesting thing in this case is that it didn't affect the whole computer like SetWindowText. It affected only the application, which eventually crashed.*

Gerzi confirmed the following Windows terminals are affected by DoS issue:

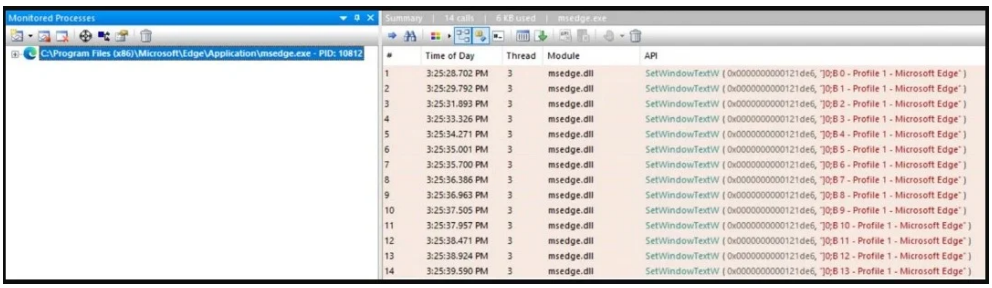
- **PuTTY** – CVE-2021-33500 (https://nvd.nist.gov/vuln/detail/CVE-2021-33500) (freezes whole computer), fixed in version 0.75
- **MobaXterm** – CVE-2021-28847 (https://nvd.nist.gov/vuln/detail/CVE-2021-28847) (freezes only app), fixed in version 21.0 preview 3
- **MinTTY** (and **Cygwin**) – CVE-2021-28848 (https://nvd.nist.gov/vuln/detail/CVE-2021-28848) (freezes whole computer), fixed in version 3.4.6
- **Git** – uses MinTTY, fixed in version 2.30.1
- **ZOC** – CVE-2021-32198 (https://nvd.nist.gov/vuln/detail/CVE-2021-32198) (freezes only app), no fix
- **XSHELL** – CVE-2021-42095 (https://nvd.nist.gov/vuln/detail/CVE-2021-42095) (freezes whole computer), fixed in version 7.0.0.76

Trying it out on web browsers

Realizing that almost all GUI applications use the SetWindowText function, the researcher tried out the attack against popular web browsers such as Chrome.

He created an HTML file that would cause the title to change rapidly in an infinite loop, forcing the browser to freeze.

The same behavior was noticed in Edge, Torch, Maxthon, Opera, and Vivaldi, all Chromium-based browsers. Though Firefox and Internet Explorer are immune to it, they still take a performance hit.

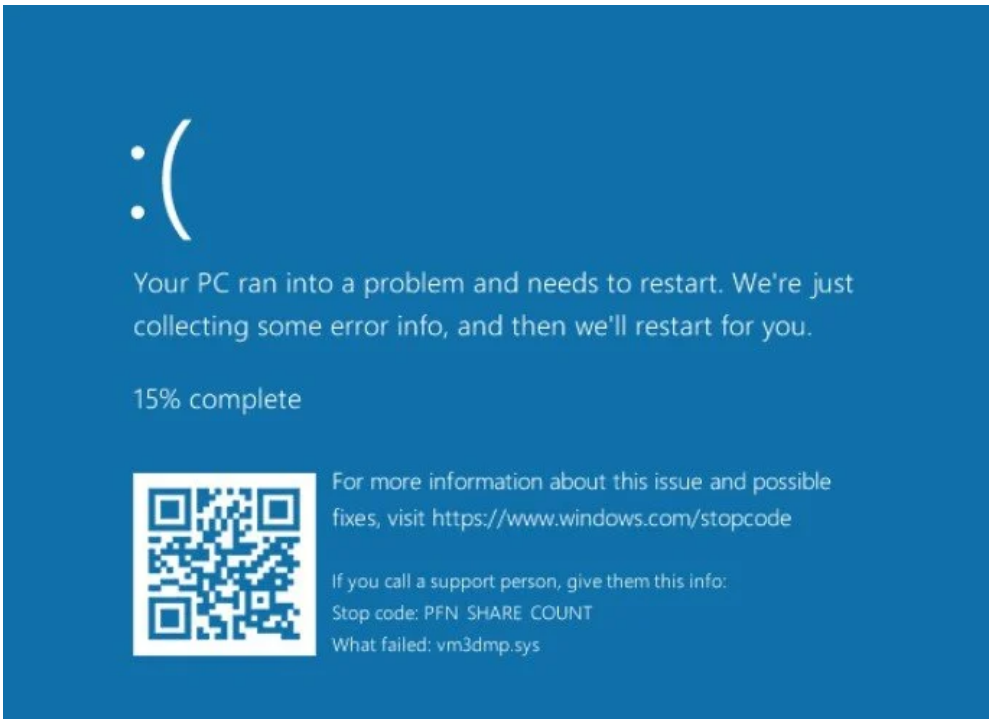


Monitoring function calls on Edge

Source: CyberArk

In all cases though, the underlying OS remains unaffected because modern browsers are based on sandboxes.

However, when trying the browser attack inside a virtual machine, a resource depletion issue occurred causing the virtualized system to display a 'Blue Screen of Death.'



BSOD when testing DoS on a virtual machine

Source: CyberArk

Response from vendors

The researcher notes that the applications affected by this attack could be anything using either SetWindowText or GdipDrawString, so the above apps are only a sample of the affected software.

Some applications like Slack, for example, feature a rate limiter on the calls of the functions, so they're resilient to this kind of DoS attacks.

Slack's limiter stopping the attack after just three calls
Source: CyberArk

Gerzi contacted the affected vendors and received the following responses:

> ***Google**: DoS issues are treated as abuse or stability issues rather than security vulnerabilities. Note: Issue is not observed on Mac but is observed on Linux. We have reviewed the issue again. We were not able to reproduce the crash in the latest versions of WS 16.1.2 build-17966106 and Chrome 92.0.4515.131. We view that the behavior you observed might be depended on chrome version used as we didn't see any BSOD issues on our end. Hence, we consider this as not a bug.*
>
> ***Vivaldi**: This is a design limitation of Windows 10; it does not limit application memory usage, and simply uses pagefile (virtual memory) when it runs out of RAM. This is slower to respond because it must be read from disk.*
>
> ***Microsoft**: Our team was able to reproduce this issue, but it does not meet our bar for servicing with an immediate security update. While this results in a denial of service condition, this can only be triggered locally and is the result of resource exhaustion. An attacker would not be able to trigger any additional vulnerable conditions or retrieve information that would be beneficial in other attacks on the system. We will be closing this case, but we have opened a bug with our development team, and they may consider addressing this in a future release of Windows.*

In response to the above, the researcher points out that it is possible to trigger the attack remotely by creating a malicious file on a remote server and opening it from a vulnerable terminal.