Multiple browsers (Microsoft Internet Explorer, Mozilla Firefox) Performance object leaks the Windows performance counter frequency (equivalent to physical CPU clock speed or virtual machine detection)

Advisory

Amit Klein

In two browser families researched (Internet Explorer and Firefox), it is possible to extract the frequency of the Windows performance counter frequency, using standard HTML and Javascript. In both browsers, window.performance.now() yields a time measurement in milliseconds (not necessarily an integral number), which is actually an integral number of the Windows performance counter ticks (i.e. 1/*f* where *f* is the Windows performance counter frequency). With multiple samples of window.performance.now() it is possible to extract this underlying time unit (e.g. using the real number version of the GCD algorithm).

With the Windows performance counter frequency, it is possible to (see details at the author's "Detecting virtualization over the web with IE9 (platform preview) and Semi-permanent computer fingerprinting and user tracking in IE9 (platform preview)" sections 5 and 6 - http://landing2.trusteer.com/sites/default/files/VM_Detection_and_Temporary_User_Tracking_in_IE9_Platform_Preview.pdf, mirror: http://dl.packetstormsecurity.net/1012-advisories/ie9-tracking.pdf)

- Remotely detect some virtual machines – by detecting two specific frequencies typically used in VM implementations, but rarely in physical machines – 14318180 Hz (HPET-based counter) and 3579545 Hz (ACPI-based counter).
- Coarse-grain fingerprint the machine - the CPU frequency is roughly 1024 times the Windows performance counter frequency. But it also seems that different machines with the same CPU clock frequency may exhibit slightly different performance counter frequencies, thus extending the fingerprint beyond simply the CPU clock speed. For example, one machine with Intel i7-3770 CPU (3.40GHz) yielded performance counter values around 3323580Hz, while another machine with Intel i7-2600 CPU (3.40GHz) yielded values around 3312805Hz (over 10000Hz apart, way beyond the measurement fluctuations). Another important feature of this fingerprinting method is that it works across the two browsers (Internet Explorer and Firefox), i.e. both browsers will produce the same fingerprint for the same machine.

The windows.performance object is supported starting Internet Explorer 10 (https://msdn.microsoft.com/en-us/library/ie/hh973355(v=vs.85).aspx) and Firefox 34 (https://developer.mozilla.org/en-US/docs/Web/API/Performance).

Proof of concept code (extracting the performance counter frequency):

```
<html>
<script>
function gcd(a,b)
{
        if (a<0.00000001)
        {
                return b;
        }
        if (a<b)
        {
                return gcd(b-Math.floor(b/a)*a,a);
        }
        else if (a==b)
        {
                return a;
        }
        else
        {
                return gcd(b,a);
        }
}

var x_init=performance.now()/1000;
var g=performance.now()/1000-x_init;
for (var i=0;i<10;i++)
{
        g=gcd(g,performance.now()/1000-x_init);
}

alert("Performance Counter Frequency: "+Math.round(1/g)+" Hz");</script>
</body>
</html>
```

Notes about the code:

- The GCD algorithm yields stable results when the initial numbers fed to it are “small”. Therefore, a baseline measurement is taken and subtracted from further measurements before being fed to the GCD algorithm.
- The counter frequency seems to be a bit unstable right after restart, and after CPU idleness. A deviation of up to 5ppm (few dozen Hz) was observed.
- The attack code was successfully tested with Internet Explorer 11 (latest at the time this advisory is written) and Firefox 37.0.1 (ditto), on Windows 8.1 64 bit (two machines) and Windows 7 SP1 64 bit (one machine). The attack succeeded for both Desktop and Metro styles of Internet Explorer (11).