FreeBSD Support for Leap Seconds

Table of Contents

1. Introduction . . . 1
2. Default Leap Second Handling on FreeBSD . . . 1
3. Cautions . . . 1
4. Testing . . . 2
5. Conclusion . . . 2

1. Introduction

A *leap second* is an one second adjustment made at specific times of year to UTC to synchronize atomic time scales with variations in the rotation of the Earth. This article describes how FreeBSD interacts with leap seconds.

As of this writing, the next leap second will occur at 2015-Jun-30 23:59:60 UTC. This leap second will occur during a business day for North and South America and the Asia/Pacific region.

Leap seconds are announced by IERS on Bulletin C.

Standard leap second behavior is described in RFC 7164. Also see time2posix(3).

2. Default Leap Second Handling on FreeBSD

The easiest way to handle leap seconds is with the POSIX time rules FreeBSD uses by default, combined with NTP. When ntpd(8) is running and the time is synchronized with upstream NTP servers that handle leap seconds correctly, the leap second will cause the system time to automatically repeat the last second of the day. No other adjustments are necessary.

If the upstream NTP servers do not handle leap seconds correctly, ntpd(8) will step the time by one second after the errant upstream server has noticed and stepped itself.

If NTP is not being used, manual adjustment of the system clock will be required after the leap second has passed.

3. Cautions

Leap seconds are inserted at the same instant all over the world: UTC midnight. In Japan that is mid-morning, in the Pacific mid-day, in the Americas late afternoon, and in Europe at night.

We believe and expect that FreeBSD, if provided correct and stable NTP service, will work as

designed during this leap second, as it did during the previous ones.

However, we caution that practically no applications have ever asked the kernel about leap seconds. Our experience is that, as designed, leap seconds are essentially a replay of the second before the leap second, and this is a surprise to most application programmers.

Other operating systems and other computers may or may not handle the leap-second the same way as FreeBSD, and systems without correct and stable NTP service will not know anything about leap seconds at all.

It is not unheard of for computers to crash because of leap seconds, and experience has shown that a large fraction of all public NTP servers might handle and announce the leap second incorrectly.

Please try to make sure nothing horrible happens because of the leap second.

4. Testing

It is possible to test whether a leap second will be used. Due to the nature of NTP, the test might work up to 24 hours before the leap second. Some major reference clock sources only announce leap seconds one hour ahead of the event. Query the NTP daemon:

```
% ntpq -c 'rv 0 leap'
```

Output that includes `leap_add_sec` indicates proper support of the leap second. Before the 24 hours leading up to the leap second, or after the leap second has passed, `leap_none` will be shown.

5. Conclusion

In practice, leap seconds are usually not a problem on FreeBSD. We hope that this overview helps clarify what to expect and how to make the leap second event proceed more smoothly.