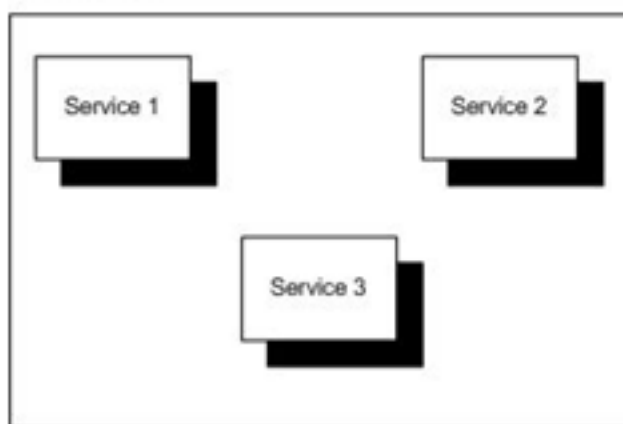
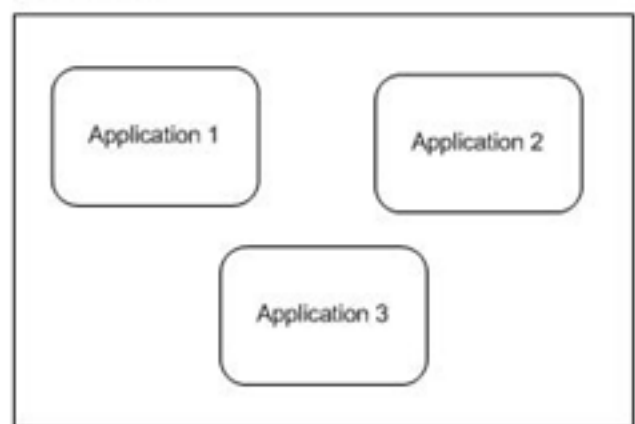


یکی از تفاوت‌های مهم امنیتی ویندوزهای جدید (از سرویس پک 2 ویندوز 2003 به بعد)، اجرای سرویس‌های ویندوز در جلسه‌ای کاری (سشنی) جدا از جلسه کاری برنامه‌های عادی است که توسط کاربران وارد شده به سیستم اجرا می‌شوند. در این ویندوزها، تنها سرویس‌ها هستند که در سشن صفر اجرا می‌شوند و مابقی برنامه‌های سایر کاربران در سشن‌های دیگر.

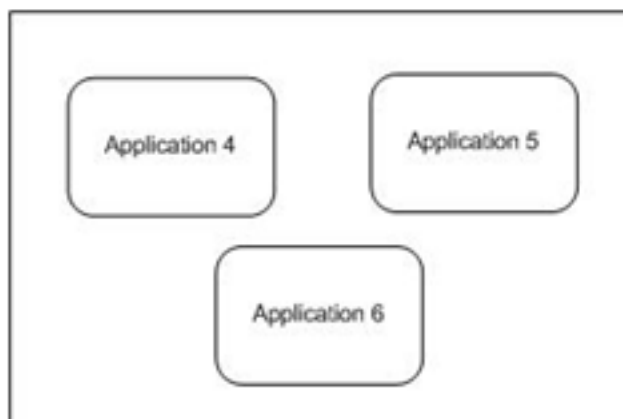
Session 0



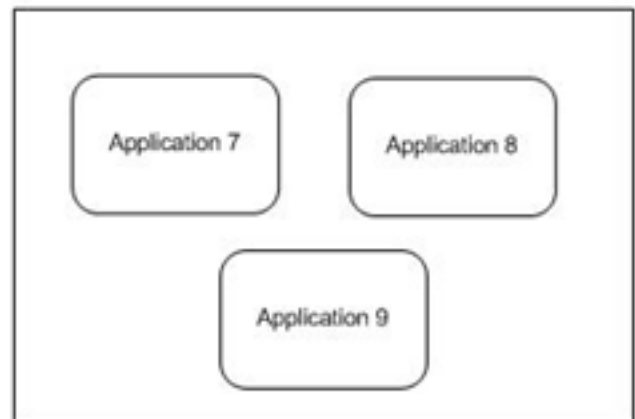
Session 1



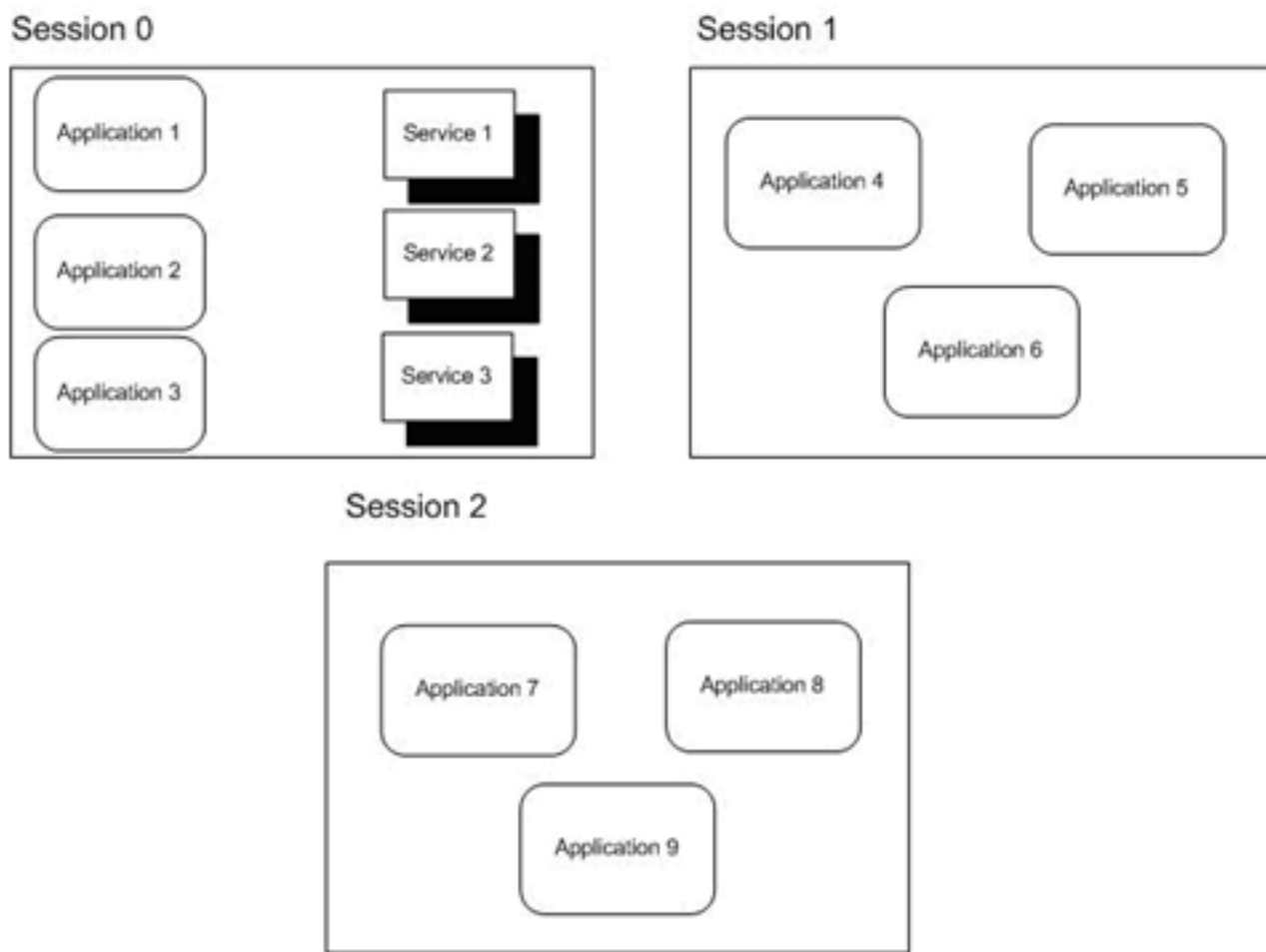
Session 2



Session 3



این برخلاف چیزی است که برای مثال در ویندوز XP وجود دارد؛ یعنی اجرای دوستانه‌ی تمام برنامه‌ها در یک سشن (چون کاربر وارد شده به سیستم نیز در سشن صفر قرار می‌گیرد).



در این حالت برنامه‌های عادی می‌توانند سرویس‌های ویندوز را به راحتی مورد حمله قرار دهند. اطلاعات بیشتر:

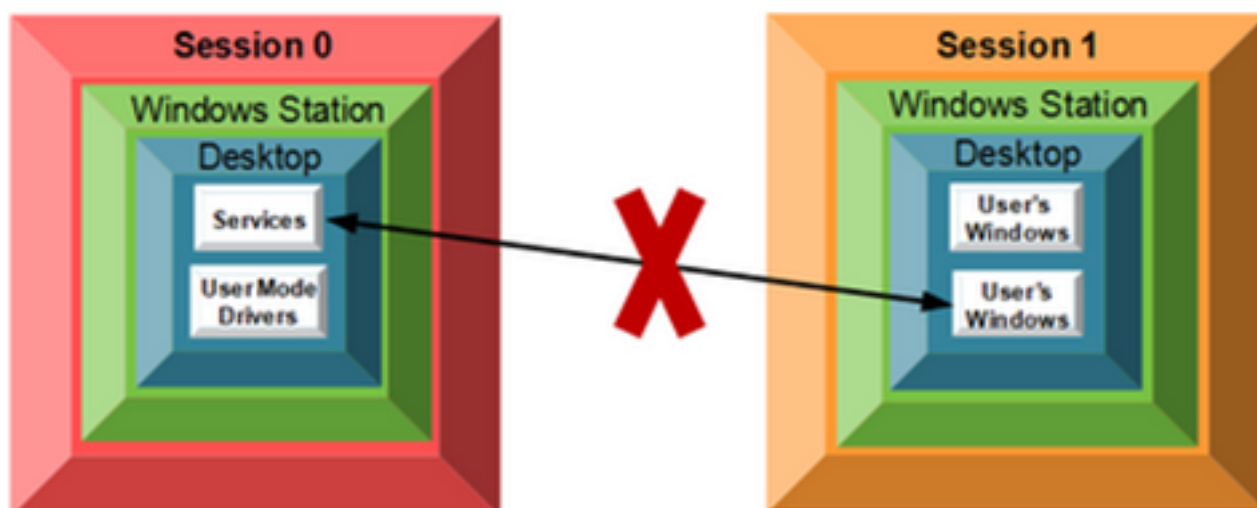
[Impact of Session 0 Isolation on Services and Drivers in Windows](#)

[Services isolation in Session 0 of Windows Vista and Longhorn Server](#)

[Session 0 Isolation](#)

[Changes to Remote Administration in Windows Server 2008](#)

مهم‌ترین معنای اجرای ایزوله سرویس‌ها در سشن صفر، به خاطر سپرده شدن سرویس‌هایی است که امکان تعامل با کاربر را داشتند؛ برای مثال سرویس‌هایی که یک رابط کاربری را نمایش می‌دادند (به کمک همان تیک معروف Allow Service to Interact with Desktop در صفحه تنظیمات یک سرویس). زیرا اکنون اولین کاربر وارد شده به سیستم، در سشن یک قرار می‌گیرد (و نه همانند قبل در سشن صفر). به این صورت دیگر به رابط کاربری نمایش داده شده در سشن صفر دسترسی نداشته و چیزی را مشاهده نخواهد کرد.



البته شاید سؤال بپرسید که چه سرویس‌هایی نیاز به نمایش رابط کاربری دارند؟ چون عموماً سرویسی که صحیح طراحی شده باشد نیاز به تعامل مستقیم با کاربر را از طریق رابط کاربری ندارد. و پاسخ این است که خیلی از سرویس‌ها! در بسیاری از برنامه‌های متداول، امکان اجرای به صورت سرویس ویندوز NT پیش‌بینی نشده است. اگر نیاز باشد تا این نوع برنامه‌ها را به صورت سرویس، برای مثال در یک ویندوز سرور اجرا کرد (تا همیشه در حال اجرا باشند و همچنین با ری استارت شدن سیستم نیز مجدداً بدون دخالت کاربر شروع به کار کنند)، در این حالت نیاز خواهد بود تا رابط کاربری آن‌ها نیز نمایش داده شوند.

سؤال: در ویندوزهای جدید برای تعامل با سشن صفر چه باید کرد؟  
در ویندوز 2003 (به همراه تمام به روز رسانی‌ها) برای اتصال به سشن صفر می‌توان دستور زیر را در خط فرمان صادر کرد:

```
Mstsc /console
```

به این ترتیب امکان اتصال به سشن کنسول یا همان سشن صفر میسر می‌شود. این پارامتر در ویندوز سرور 2008 ندید گرفته خواهد شد زیرا در این ویندوز امکان نصب یک سرویس در سشن‌هایی بجز سشن صفر نیز پیش‌بینی شده است؛ هر چند پیش فرض نصب همان سشن صفر است. اما به جهت سازگاری با نگارش‌های قبل این بار بجای پارامتر console از پارامتر admin [استفاده کنید](#).

با استفاده از برنامه نویسی چطور؟  
روش کار [در این مقاله](#) توضیح داده شده است اما سطر اول آن به این نکته تأکید دارد که ... این کار را نکنید! سرویسی که درست طراحی شده باشد نیازی به رابط کاربری تعاملی عموماً جهت اعمال تنظیمات آن، ندارد.

## اصلاح daylight saving time ویندوز تا 90 سال بعد

عنوان:

وحید نصیری

نویسنده:

۰۳۱ ۱۳۹۱/۰۶/۳۱

تاریخ:

[www.dotnettips.info](http://www.dotnettips.info)

آدرس:

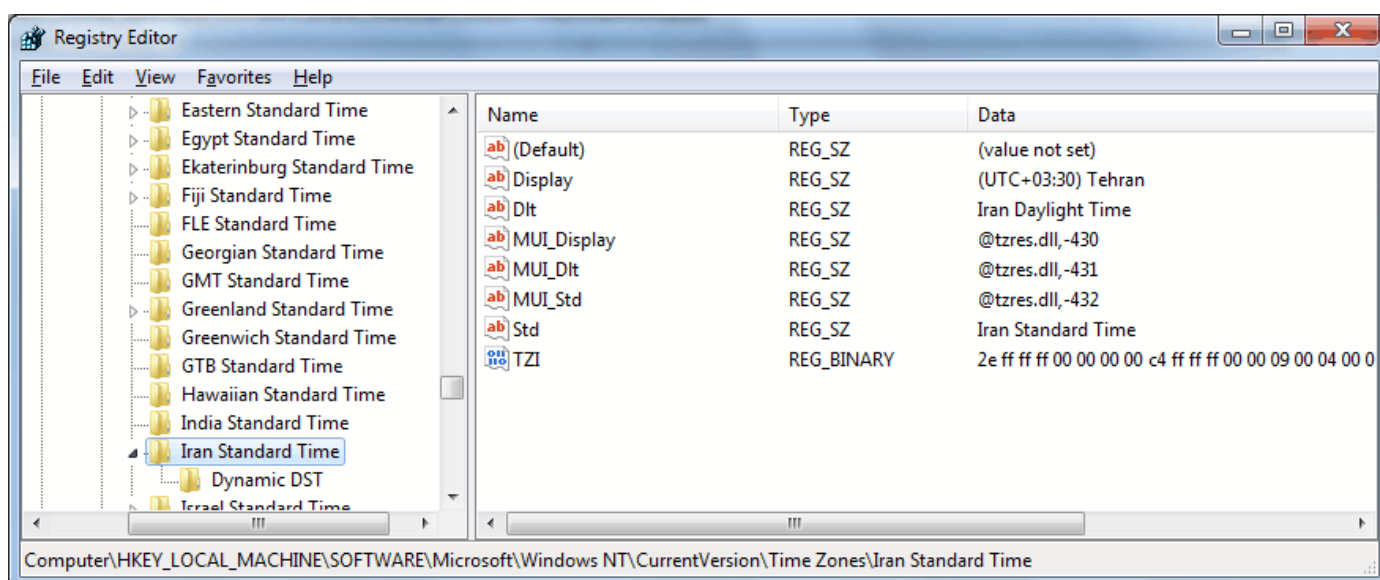
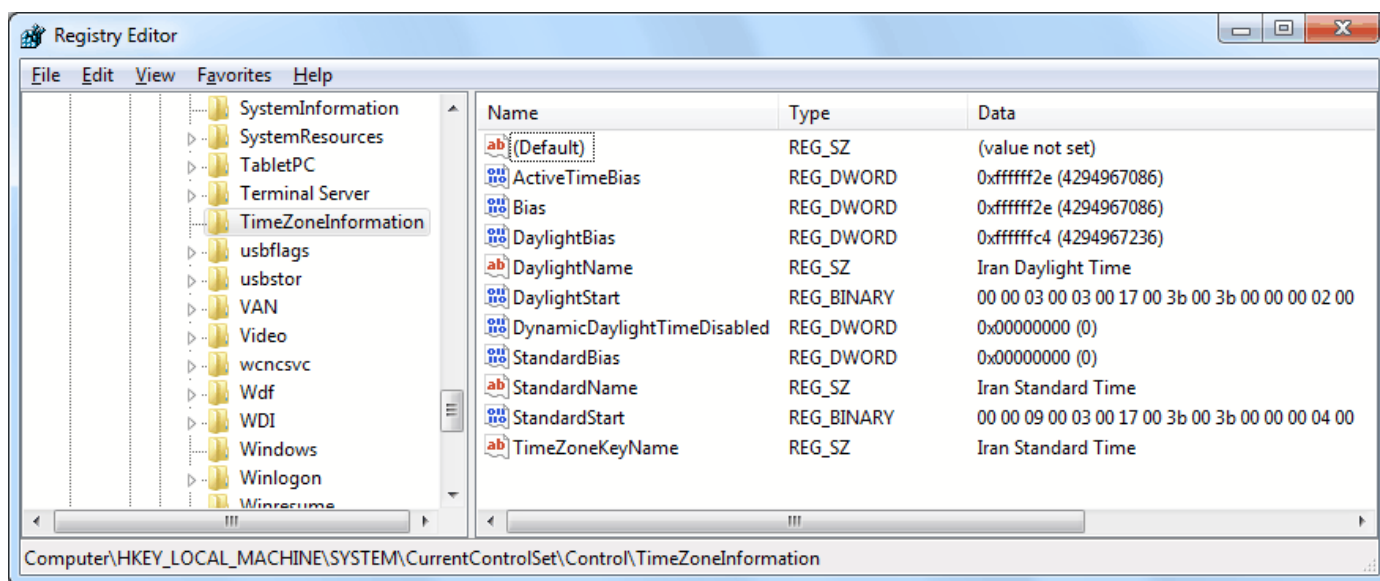
Windows 7, Windows 8

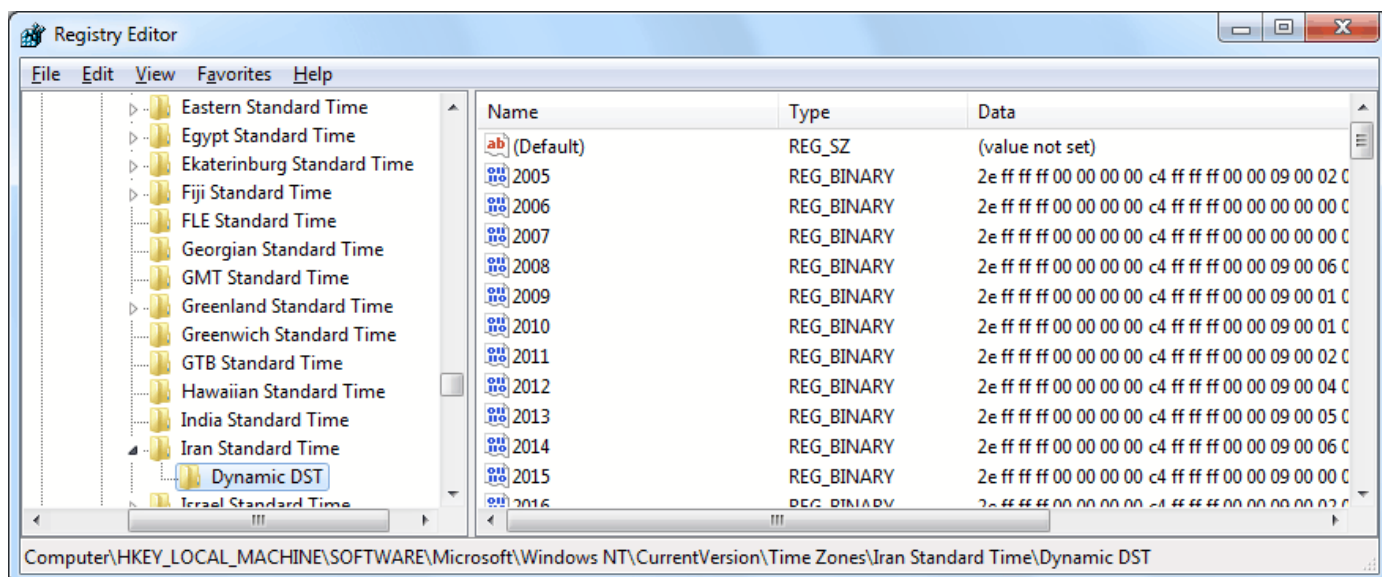
گروه‌ها:

چند سالی هست (از سال 2009) که آپدیت‌های daylight saving time ویندوز شامل حال تنظیمات رسمی ایران نمی‌شود. برای نمونه، همین یکی دو روز قبل بود که ساعت ویندوز به صورت خودکار تغییر کرد؛ درحالی‌که باید در انتهای روز 30 شهریور اینکار صورت می‌گرفت.

اطلاعات daylight saving time یا بازه صرفه جویی زمانی ویندوز در دو مدخل رجیستری زیر ثبت می‌شوند:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation]
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones\Iran Standard Time]
```





تصویر سوم مرتبط است به ویندوزهای ویستا به بعد که مفهوم dynamic daylight saving time در آن‌ها معرفی شده است. در اینجا یک نمونه اطلاعات زمانی ثبت شده مرتبط با ایران را مشاهده می‌کنید:

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones\Iran Standard Time]
"Display"="(GMT+03:30) Tehran"
"Dlt"="Iran Daylight Time"
"Std"="Iran Standard Time"
"MapID"="-1,72"
"Index"=dword:000000a0
"TZI"=hex:2e,ff,ff,ff,00,00,00,00,c4,ff,ff,ff,00,00,09,00,04,00,03,00,17,00,3b,\
00,3b,00,00,00,00,00,00,03,00,02,00,03,00,17,00,3b,00,3b,00,00,00
```

TZI ایی که در اینجا وجود دارد، دارای یک چنین ساختاری است:

```
using System.Runtime.InteropServices;

namespace TimeZoneInfo.Core
{
    [StructLayout(LayoutKind.Sequential)]
    public struct TZI
    {
        public int Bias;
        public int StandardBias;
        public int DaylightBias;
        public SystemTime StandardDate;
        public SystemTime DaylightDate;
    }
}
```

و SystemTime آن نیز به نحو زیر تعریف شده است:

```
using System;
using System.Runtime.InteropServices;

namespace TimeZoneInfo.Core
{
    [StructLayoutAttribute(LayoutKind.Sequential)]
    public struct SystemTime
    {
        public short Year;
        public short Month;
        public short DayOfWeek;
        public short Day;
```

```

    public short Hour;
    public short Minute;
    public short Second;
    public short Milliseconds;
}
}

```

برای مثال اگر اطلاعات درج شده در TZI به صورت زیر باشد:

```

2C 01 00 00 00 00 00 00
C4 FF FF FF 00 00 0A 00
00 00 05 00 02 00 00 00
00 00 00 00 00 00 04 00
00 00 01 00 02 00 00 00
00 00 00 00

```

نمونه رمزگشایی شده آن به نحو ذیل خواهد بود:

```

(little-endian) => (big-endian)
2C 01 00 00 => 00 00 01 2C = 300 Bias
00 00 00 00 => 00 00 00 00 = 0 Std Bias
C4 FF FF FF => FF FF FF C4 = 4294967236 Dlt Bias
( SYSTEM TIME ) StandardDate
00 00 => 00 00 = Year
0A 00 => 00 0A = Month
00 00 => 00 00 = Day of Week
05 00 => 00 05 = Day
02 00 => 00 02 = Hour
00 00 => 00 00 = Minutes
00 00 => 00 00 = Seconds
00 00 => 00 00 = Milliseconds
( SYSTEM TIME ) DaylightDate
00 00 => 00 00 = Year
04 00 => 00 04 = Month
00 00 => 00 00 = Day of Week
01 00 => 00 01 = Day
02 00 => 00 02 = Hour
00 00 => 00 00 = Minutes
00 00 => 00 00 = Seconds
00 00 => 00 00 = Milliseconds

```

در ساختار SystemTime متناظر با TZI، فیلد Day دارای مقدار روز در یک ماه نیست. به معنای شماره هفته است. مثلاً پنج شنبه (DayOfWeek) هفته سوم (Day) ماه 9 سال 2012.

همچنین Day از یک شروع می‌شود و DayOfWeek از صفر. Year اگر صفر وارد شود به معنای زمان نسبی است و برای سال بعد نیز می‌تواند کاربرد داشته باشد (و عموماً صفر تعریف شده است).

بنابراین برای تبدیل DateTime به SystemTime سازگار با TZI به فرمول زیر خواهیم رسید:

```

public static SystemTime ToSystemTime(DateTime time)
{
    var result = new SystemTime
    {
        Year = 0, // سال نسبی وارد می‌شود نه مطلق
        Month = (short)time.Month,
        DayOfWeek = (short)time.DayOfWeek,
        Hour = (short)time.Hour,
        Minute = (short)time.Minute,
        Second = (short)time.Second,
        Milliseconds = (short)time.Millisecond
    };

    int weekdayOfMonth = 1; // شماره هفته است نه شماره روز
    for (int dd = time.Day; dd > 7; dd -= 7)
        weekdayOfMonth++;

    result.Day = (short)weekdayOfMonth;

    return result;
}

```

در ادامه نیاز خواهیم داشت تا ساختار TZI سفارشی و بازسازی شده خودمان را بتوانیم به آرایه‌ای از بایت‌ها تبدیل کنیم تا بتوان در همان مدخل رجیستری نوشت. اینکار را توسط متد `SerializeByteArray` زیر می‌توان انجام داد:

```
using System;
using System.Runtime.InteropServices;

namespace TimeZoneInfo.Core
{
    public static class ByteUtils
    {
        public static Byte[] SerializeByteArray<T>(T msg) where T : struct
        {
            int objsize = Marshal.SizeOf(typeof(T));
            Byte[] ret = new Byte[objsize];

            IntPtr buff = Marshal.AllocHGlobal(objsize);
            Marshal.StructureToPtr(msg, buff, true);
            Marshal.Copy(buff, ret, 0, objsize);
            Marshal.FreeHGlobal(buff);

            return ret;
        }
    }
}
```

و اگر اینکار را تا بیش از 90 سال بعد بر اساس تاریخ ایران انجام داده و مداخل رجیستری ویندوز را تکمیل کنیم، خروجی آن فایل `reg` زیر خواهد بود که به سادگی با کلیک راست و انتخاب گزینه‌ی `merge` به رجیستری ویندوز اضافه شده و تا چندین سال بعد، مشکل تنظیمات DST را برطرف خواهد کرد:

[win7-8-dst-patch.zip](#)

پس از اعمال تغییرات فوق، نیاز است یکبار ویندوز را ری استارت کنید.

## نظرات خوانندگان

نویسنده: جلال  
تاریخ: ۲۱:۴۲ ۱۳۹۱/۰۶/۳۱

البته برای ریستارت عجله نکنید، چون 6 ماه وقت دارید :

نویسنده: محمد حسین عامری  
تاریخ: ۱۴:۱۹ ۱۳۹۲/۱۲/۲۷


با سلام  
متأسفانه برنامه شما خروجی اشتباه تولید می‌کند برای مثال زمان تغییر ساعت نوروز 93 ، March/22/2014 می‌باشد اما برنامه شما تاریخ March/21/2014 را تولید میکند.  
با احترام  
محمد حسین عامری

نویسنده: وحید نصیری  
تاریخ: ۱۴:۲۴ ۱۳۹۲/۱۲/۲۷

خیر. به ساعت آن هم باید دقت کنید. تنظیم شده به 23:59 دقیقه، روز 21 مارس:

سه شنبه ۲۷ اسفند ۱۳۹۲

مارس، ۲۰۱۴						
ج	پ	چ	س	د	ی	ش
۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲
۷	۶	۵	۴	۳	۲	۱
۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸
۲۱	۲۰	۱۹	۱۸	۱۷	۱۶	۱۵
۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲
۴	۳	۲	۱	۳۱	۳۰	۲۹



ب.ظ ۱۰:۲۲:۱۰  
سه شنبه

Daylight Saving Time begins on at ۱۳۹۳ فروردین ۱  
۱۱:۵۹ The clock is set to go forward 1 hour at that  
time.

ب.ظ

[Change date and time settings...](#)

نویسنده: مصطفی



تاریخ: ۱۳:۵۰ ۱۳۹۳/۰۱/۰۲

خیلی ممنون مشکل من رو هم برطرف کرد. برای تهران daylight روی ویندوز ویستا تعریف نشده بود. اگر میشه یه چندتا جمله دیگه به آخر متنتون اضافه کنید که داخل جستجوگرها لینک شما نشون داده بشه. مثلاً

problem Daylight Saving Time not Defined for Tehran Iran

Vista Daylight Saving Time for Teharn

Vista Automatically adjust clock for Daylight Saving Time for Teharn

Fix Automatically adjust clock for Daylight Saving Time for Teharn

نویسنده: سمیعی، محمدصادق

تاریخ: ۱۲:۴۹ ۱۳۹۳/۰۱/۰۵

این patch رو در ویندوز سرور 2003 تست کردم اما برای نوروز 93 درست کار نکرد.

نویسنده: وحید نصیری

تاریخ: ۱۳:۱۳ ۱۳۹۳/۰۱/۰۵

بله. ویندوز XP و ویندوز سرور 2003 دارای daylight saving time **dynamic** توضیح داده شده در بحث نیستند. patch ارائه شده به کمک dynamic daylight saving time سال‌های مختلف را اضافه کرده و این مورد از ویندوز ویستا به بعد به ساختار ویندوز اضافه شده‌است. برای ویندوزهای XP و 2003 از [برنامه TZEEdit](#) مایکروسافت استفاده کنید (هر سال هم باید یکبار انجام شود؛ برخلاف ویندوزهای ویستا به بعد که با استفاده از روش dynamic معرفی شده فقط یکبار نیاز به تنظیم دارند).

نویسنده: حامد قنادی

تاریخ: ۸:۱۹ ۱۳۹۳/۰۱/۰۶

درود بر شما

آیا در ویندوز سرور 2008 و 2012 هم جواب می‌دهد؟

نویسنده: وحید نصیری

تاریخ: ۸:۴۳ ۱۳۹۳/۰۱/۰۶

بله. فایل [win7-8-dst-patch.zip](#) را برای ویندوزهای جدید اجرا کنید.

در حین نصب ویژوال استودیو 2012 بر روی ویندوز 7 تازه که بصورت virtual هاست شده بوده خطایی مبنی بر عدم وجود ریشه گواهی Certificate ها در سیستم اتفاق افتاد که مانع از نصب بعضی از پکیج‌های نصاب گردید.

```
1. http://stackoverflow.com/questions/16673292/microsoft-web-deploy-3-0-a-certificate-chain-could-not-be-built-to-a-trusted-r
2. http://blogs.msdn.com/b/heaths/archive/2012/08/17/a-certificate-chain-could-not-be-built-to-a-trusted-root-authority.aspx
3. http://social.msdn.microsoft.com/Forums/vstudio/en-US/aeb3a43d-e5d7-41ab-b875-6a0d3b438abf/web-deploy-3-setup-problem
4. http://msdnrss.thecoderblogs.com/2012/08/a-certificate-chain-could-not-be-built-to-a-trusted-root-authority/
5.
http://forums.asp.net/p/1897741/5361492.aspx?Microsoft+Web+Deploy+3+0+A+certificate+chain+could+not+be+built+to+a+trusted+root+authority+
```

مشکل نیز با توجه به توضیح مشخص بود. راه حل‌ها نیز ختم به نصب یک آپدیت [KB2746268](http://kb2746268) برای ثبت گواهی‌ها می‌شد که برای نصب این آپدیت میبایست ویندوز باصطلاح Genuine باشد و بعد از نصب نیز باید نصاب ویژوال استودیو را دوباره راه انداخت و گزینه‌های نصب نشده را دوباره انتخاب کرد. ولی با توجه به تجربه شخصی از محصولات مایکروسافت در حین نصب ویژوال استودیو ارتباط با اینترنت را برای ویندوز مجازی میسر کردم و خوشبختانه نتیجه مورد نظر حاصل شد و پس از طی مراحل Add\Remove مربوط به قابلیت‌های نصب نشده ویژوال استودیو تمامی پکیج‌ها بصورت صحیح و بدون مشکل نصب شدند.