دراسة مفصلة في Windows Registry

في نظر البعض قد تُعتبر الريجستري منطقة محظورة بينما يعتبر البعض الآخر أن الريجستري مكان من الممكن تغير الكثير من الإعدادات فيه لما يناسب الذوق ومتطلبات العمل. سنحاول هنا بعونه تعالى تبسيط مفهوم الريجستري إلى الحد الذي يجعلها سهلة الهضم للغالبية العظمى من مستخدمي الحاسوب إن كانت لهم العزيمة والهمة للقراءة والتفاعل البناء والصبر. فموضوع الريجستري موضوع مهم جداً ولابد لكل مستخدم كمبيوتر من أن يمتلك ولو على الأقل بعض أساسيات الريجستري إن لم نقل أن يصبح محترفاً بخباياها وأسرارها. ونظراً لكون المدخل إلى الموضوع سيكون خطوة خطوة فسيأخذ بعض الحيز من الوقت وسيكون بكل تأكيد نوعاً ما مطولاً، سأقوم إن شاء الله بتقسيمه على مراحل الواحدة تلو الأخرى. ما أرجوه (إن كان ذلك ممكناً بكل تأكيد) من إدارة المنتدى الموقرة دراسة جدوى تثبيت هذا الموضوع ولو لفترة معينة أو لحين الإنتهاء من تكملته لغرض أن تعم الفائدة ولكم مني مقدماً جزيل الشكر والتقدير وبارك الله فيكم وفي سعيكم إلى الخير.

ما هي ريجستري الويندوز؟

تعتبر الريجستري أحد أهم ركائز نظام التشغيل (الويندوز)، لكونها تمثل قاعدة البيانات database التي يستخدمها الويندوز لخزن المعلومات التي تخص

إعدادات وأماكن وجود البرامج ومكونات الويندوز على القرص الصلب. معظم البرامج أثناء تنصيبها تكتب إلى الريجستري معلومات عن إعداداتها وأرقام إصداراتها وأمكن تواجدها وغيرها على القرص الصلب. قد يختلف مسار المعلومات المدونة في الريجستري من ريجستري إلى أخرى باختلاف نسخ الويندوز، وكذلك لكل ويندوز وجهاز ريجستري خاصة بهما.

مما يتكون أساس الريجستري؟
يتكون الهرم التسلسلي لريجستري الويندوز من ما يلي:
1 - الشجرة Tree
2- إلتقاء التقاطع أو العقد Nodes
3- المفاتيح Keys

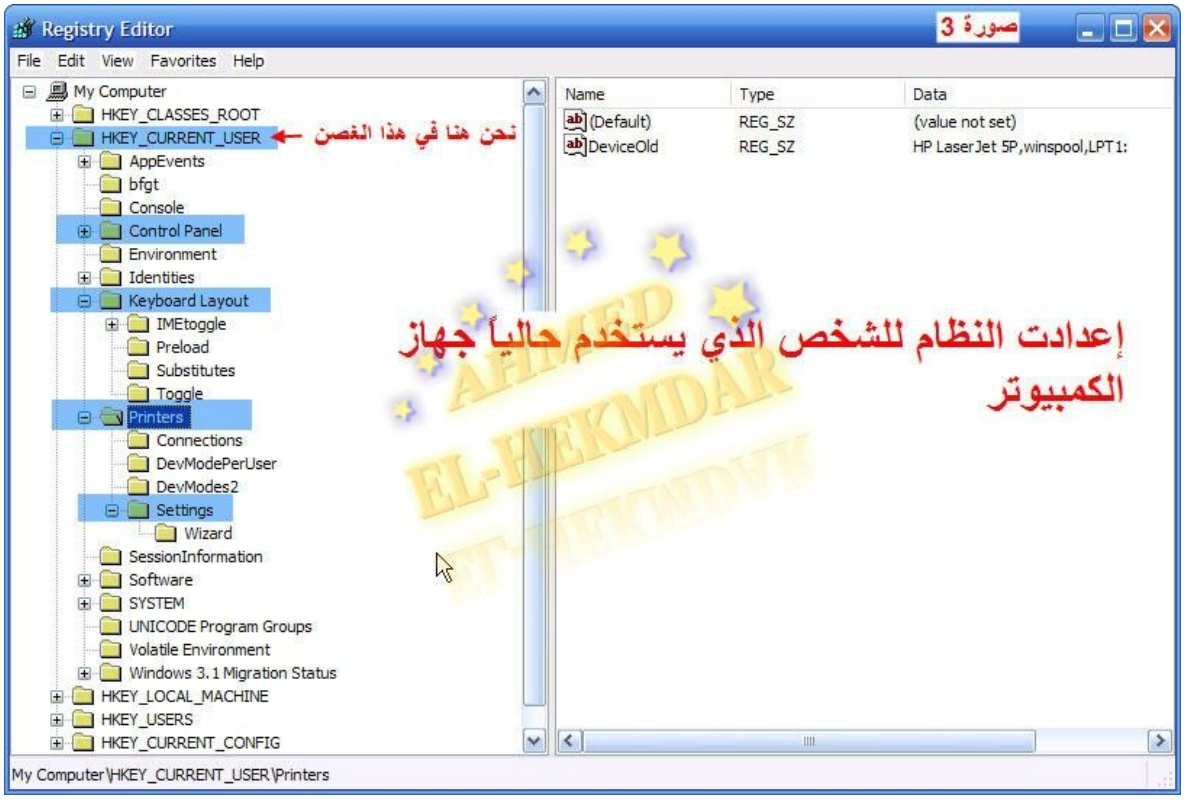
ولتبسيط الأمر للقارئ العربي الكريم، نتخيل الريجستري كشجرة، مكونة من ساق وغصون أو أغصان بحيث يكون الحد الأعلى لعدد الغصون المنبثقة من الساق بستة وغالباً ما تكون أربعة أو خمسة ولكن لا تكون بأي حال من الأحوال أقل من أربعة غصون وسنعلم السبب لا حقاً إن شاء الله تعالى.. نقطة إلتقاء الغصن بالساق تسمى بالعقدة Node ولكل عقدة مفتاح (Keyعنوان أو إسم) من دون أية قيمة أنظر إلى الصورة 1. وبما أن الغصن الواحد ممكن أن يتفرع إلى فروع رئيسية ثم إلى فروع جانبية ...إلخ بحيث تصبح التفرعات متشعبة ومتشابكة إلى الحد الذي يمكن إعتبار الغصن الواحد كخلية نحل لذلك أطلق إسم Hive على الغصن الواحد.

وبنفس الطريقة التي ارتبط الساق بالغصن بعقدة ومفتاح يرتبط الفرع الرئيسي بالغصن بعقدة ومفتاح ولكن هنا مع إمكانية إعطاء قيمة للمفتاح وممكن أن لا يمتلك المفتاح هنا على قيمة أيضاً. كما يرتبط الفرع الجانبي مع الفرع الرئيسي للغصن الواحد أيضاً بعقدة ومفتاح وأيضاً للمفتاح هنا على الأقل له قيمة واحدة أو أكثر وقد لا يمتلك قيمة. وهكذا يستمر التفرع بنفس الطريقة إلى أن يصل إلى الثمرة أو مفتاح الشفرة وعندها يجب أن تكون هناك قيمة أو عدت قيم أنظر إلى الصورة 2. ومما تجدر الإشارة إليه هو أنه ليس هناك أي تشابك أو إتصال مباشر بين غصن وآخر بل لكل غصن تفرعاته وثماره المستقلة عن الغصن الآخر.

HKEY_CURRENT_USER

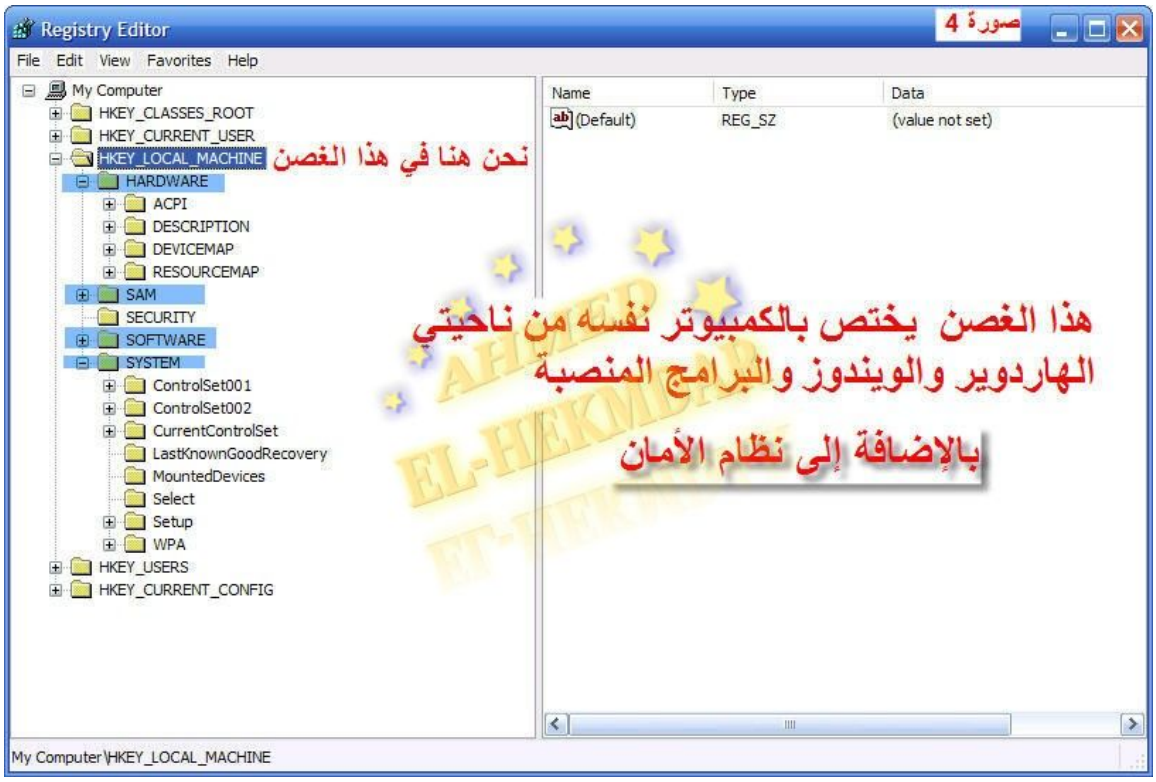
هذا الغصن يحوي على معلومات معايرة configuration لإعدادات النظام الخاصة بالشخص الذي يستخدم الويندوز حالياً. ومن هذا جاء إسم هذا الغصن أي

أن user تعني المستخدم أو الشخص المستخدم للحاسوب وكلمة current تعني حاليا أوفي الوقت الحاضر. وعليه تكون وظيفة هذا الغصن هي التحكم بإعدادات مستخدم الكمبيوتر حالياً لمثلاً سطح المكتب Desktop، مظهر الشاشة، شاشة التوقف، عمق ألوان الشاشة، إعدادات الإنترنت وطريقة الإتصال بالعالم الخارجي، إعدادات الأجهزة المرتبطة بالجهاز من طابعة وسكانر وغيرها (مع ملاحظة أن هذا لا يعني إحتواء هذا الغصن على معلومات عن مشغلات الأجهزة والتي سنتناولها في الغصن القادم)، كما يحتوي هذا الغصن على معلومات إعدادات عن نظام الأمان وبرامج الحماية المستخدمة من قبل المستخدم الحالي للجهاز. فمثلاً، عندما تقوم بتغير شاشة التوقف، فإن إسم شاشة التوقف القديمة سيتم إستبداله حالاً في هذا الغصن بإسم شاشة التوقف الجديدة وبذلك يتم الإحتفاظ به إلا أن يتم تغيير آخر وهكذا مع كل تغيير يقوم به مستخدم الحاسوب الحالي. وعند كل إقلاع للويندوز، يذهب الويندوز إلى هذا الغصن قبل ظهور سطح المكتب لقراءة الإعدادات الخاصة لذلك المستخدم الحالي ومن ثم تطبيقها ولهذا لا تتغير الإعدادات إلا أن يقوم المستخدم بتغييرها بإرادته. وملخصاً أن هذا الغصن يحوي فقط على معلومات معايرة لإعدادات النظام للشخص الذي يستخدم الجهاز حالاً.



HKEY_LOCAL_MACHINE

هذا الغصن يحتوي على معلومات تخص الكمبيوتر نفسه من ناحيتي الهاردوير ونظام التشغيل (الويندوز) والبرامج المنصبة. أي بمعنى أن هذا الغصن يحوي على معلومات مفصلة ومرتبة بنفس الوقت عن كل قطعة من قطع الهاردوير في الجهاز، من لوحة مفاتيح، ماوس، نوافذ الطابعة printer ports، أقراص التخزين الصلبة، سيدي وديفيدي وسيدي رايتر ...إلخ. كما يحوي هذا الغصن على مشغلات الأجهزة drivers، وعلى قائمة بكل البرامج المنصبة وملفاتها التابعة لها، كما أن هذا الغصن مسؤول عن البرامج التلقائية التشغيل مع بداية كل إقلاع للويندوز



ولهذا الغصن المتشعب على الأغلب خمسة فروع رئيسية هي:

1- الهاردوير Hardware ووظيفته الخزن المرحلي إن صح التعبير لإعدادات مشغلات الأجهزة Device drivers وكذلك أرقام IRQ وغيرها. في كل مرة يشغل أو يعاد تشغيل الجهاز يعاد تكوين هذا الفرع الرئيسي. قد يتسائل البعض لماذا يعاد تكوين هذا الفرع الرئيسي في كل مرة يشتغل أو يعاد تشغيل الجهاز فيها: أقول المسألة منطقية تماماً، فلو لم تكن كذلك فكيف سيتحسس النظام بأي تغيير سواء

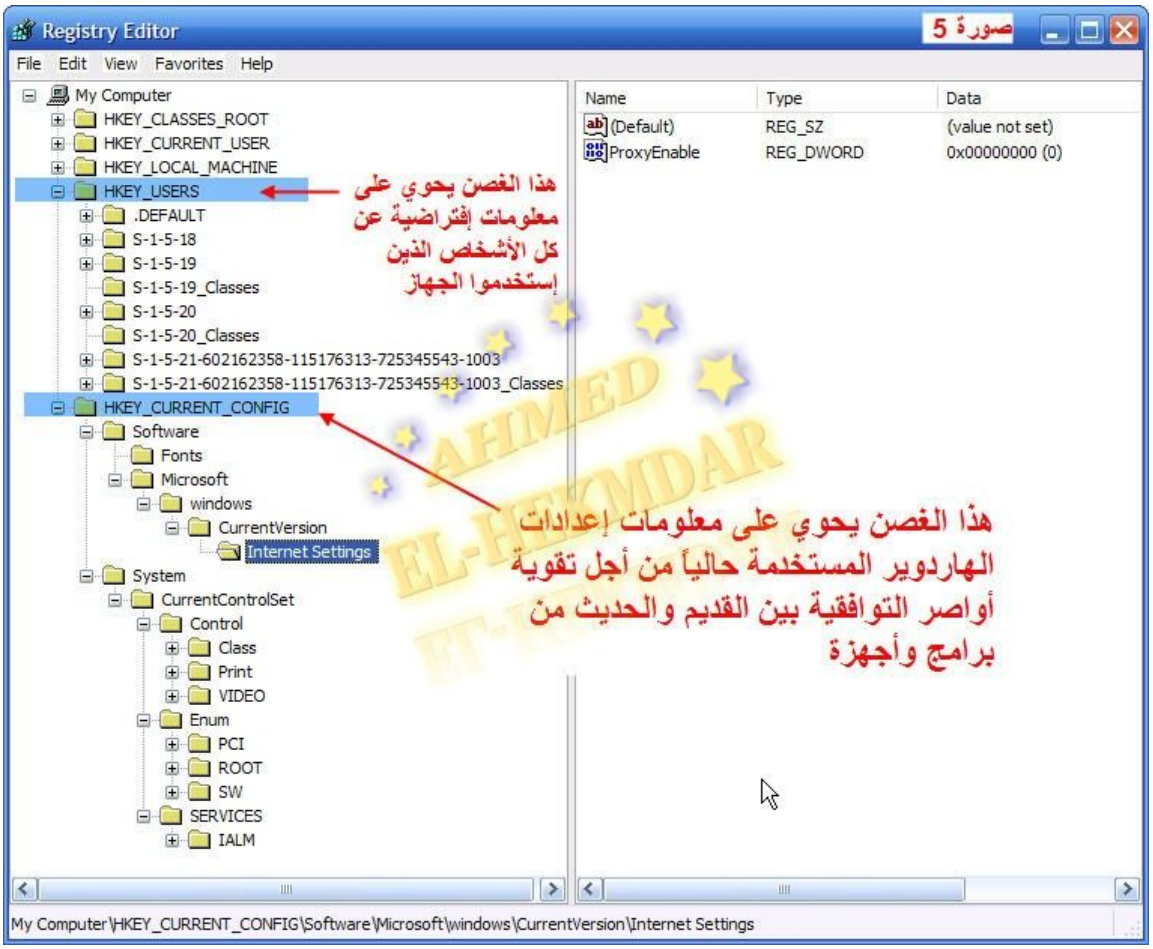
أكانت في ملفات التشغيل أو الأجهزة أو قطع الهاردوير المرتبطة بالكمبيوتر.
2-إدارة التحكم بنظام الأمان وإشتراكات المستخدمين Security accounts manager (SAM): هذا الفرع الرئيسي وكما يستدل من مفاتحه (عنوانه) أن وظيفته خزن المعلومات الخاصة بنظام الأمان للجهاز security settings، وخزن المعلومات الخاصة بحسابات أو إشتراكات أو عضويات مستخدمي الحاسوب سواء أكانوا أفراد أو مجموعات أو شركات user accounts and group memberships .
3-الأمان Security: وهذا الفرع الرئيسي وظيفته البحتة التحكم بمصير مستخدم الكمبيوتر من حيث حقه في الدخول وإستخدام الجهاز، حيث يتحكم بكلمة العبور للمستخدم إن وجدت، فإن كانت صحيحة سمح له الغصن بالدخول وإستخدام الجهاز وإلا كلا.
4-البرامج Software: حيث يحتوي هذا الفرع الرئيسي على كل ما يخص البرامج من عناوين إلى أماكن تواجد إلى تواريخ ومصادر الإنشاء ... ألخ. ومما تجدر الإشارة إليه هنا هو أن فرع البرامج هذا لا يرتبط بمستخدم أي بمعنى انه عام، بمعنى آخر أي شخص يسمح له فرع الأمان بالدخول يستطيع إستخدام البرامج بالرغم وكما قلنا أعلاه أنه لا توجد أي علاقة مباشرة أو رابط مباشر بين فرع وفرع بل توجد علاقة غير مباشرة بين فرع وفرع، وهذا هو أحد العلاقات الغير المباشرة.
5-النظام System: هذا الفرع الرئيسي يحوي على كل ما له علاقة بإقلاع النظام وإغلاقه. فأغلب التحسينات التي يمكن إجراءها لتسريع عمليتي الإقلاع والإغلاق يجب أن تتم هنا.

ملاحظة هامة: الفروع الرئيسية الثلاثة الأولى أي Hardware، SAM و Security لا يمكن إجراء تعديل على قيمها هنا داخل هذا الغصن، بينما يسمح الفرعين الرئيسيين الأخيرين اي Software و system بالسماح لإجراء التغيرات على قيمهما هنا.

الغصن المتشعب HKEY_USERS

بإختصار شديد هذا الغصن يحوي على معلومات إفتراضية عن كل الأشخاص الذين إستخدموا الجهاز فيما لو كان هناك عدت اشخاص يستخدمون الجهاز الواحد في

أوقات مختلفة. أهمية هذا الغصن تبدو جلية في مقاهي الإنترنت والكليات والجامعات حيث يكثر إستخدام الجهاز الواحد من قبل أكثر من رجل. اما بالنسبة للشخص الواحد الذي يستخدم جهازه لوحده فلهذا الغصن أهمية قليلة جداً ولكن لا يمكن مسحه أو حذفه.



نحن نتكلم بشكل عام عن موضوع ريجستري الويندوز، وكما قلنا في بداية الموضوع بأن للريجستري مفاتيح وقيم عددية أو رقمية. قيم المفاتيح هذه في الريجستري إما أن تكون بنظام الأرقام العشرية الذي نعرفه منذ السنوات الأول من أعمارنا، وإما بنظام الأرقام هكس الذي هو للأساس 16. فأهمية أنظمة الأرقام هذه ليس فقط في ريجستري الويندوز وإنما أيضاً في البرمجة.

فلو لم أكن قد أعطيت هنا النبذة المبسطة عن نظم الأرقام هذه لأصبح موضوع ريجستري الويندوز وبخاصة عند إدخال قيم المفاتيح صعباً. أرجو منك قراءة الموضوع أكثر من مرة حتى تصبح لك الفكرة واضحة عن موضوع الأرقام. فهو مهم ويبني قاعدة أساسية متينة لفهم ريجستري الويندوز.

نبذة عن أنظمة الأرقام Decimal و Hexadecimal

قد يتساءل البعض ما هو أصل الكلمة digit؟ أقول أصلها قادم من الكلمة اللاتينية digitus وتعني الأصبع. وبمرور الزمن ونظراً للحاجة الماسة إليه، تطور مفهوم الحساب لدا الناس بحيث أخذوا يشيرون إلى الأصابع أو digits على أنها أرقام numbers. ثم أخذوا يتعلمون العد بإستخدام الأصابع، ومن هنا يُعتَقَد أن أصل النظام العشري Decimal الذي نألفه كثيراً قادم من فكرة الأصابع العشرة! فنظام Decimal أو ما يسمى بنظلم الأرقام للأساس 10، هو نفس النظام الذي نبدأ مشوارنا منذ الصغر في تعلمه، فهو يبدأ من الصفر كأول رقم وينتهي ب 9 ثم يعيد هذا النظام نفسه إبتداءاً من 10 ...19 ثم يعيد نفسه تارة أخرى وهكذا يستمر بنفس المنوال.
يعتبر Hexadecimal أو ما يعرف مختصراً ب Hex من أنظمة الأرقام الغير المسموع بها كثيراً خارج عالم برمجة الكمبيوتر. ولكن في الحقيقة يعتبر من أهم أنظمة الأرقام. فلو أردت تعلم لغة البرمجة C أو assembly وحتى QB على ما أعتقد، فلن تصبح عملية التعلم سهلة إلا بإمتلاك إلمام كاف بنظام الأرقام Hex. وكما لنظام Decimal الأساس 10 فإن لنظام Hexadecimal

الأساس 16: ولكن ما هو أصل الأساس 16؟ فلو قمنا بتجزئة كلمة Hexadecimal إلى Hexa والذي يمثل رقم 6 في اللاتيني و decimal والذي يمثل رقم عشرة لأصبح أصل الأساس جلياً لنا.

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Hexadecimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10

لو تأملنا الجدول أعلاه، لرأينا أن النظامين متشابهين تماماً إلى حد ومن ضمنها الرقم 9، بعد ذلك يبدأ الإختلاف. فمثلما يعيد نظام Decimal نفسه من 10 فإن نظام Hex يعيد نفسه من 16 وهكذا. وكما ترون فإن الأرقام من 10 إلى 15 تمثل في نظام Hex بالأحرف من A إلى F، أي بدلاً من أن نتعامل مع رقم مكون من 2 digits سنتعامل في نظام Hex بحرف. قد يتساءل البعض لماذا هذا التعقيد؟ لماذا لا يتم البقاء على نظام Decimal الذي نألفه، أقول فائدة نظام Hex ستبدو أكثر جلاءاً عندما نتعامل مع الأرقام الكبيرة جداً حيث سيتم التعامل مع عدد من الأحرف والأرقام مجتمعين معاً بدلاً من رقم طويل مكون عدد كبير من ال digits.

أمثلة:

نقوم بتحويل الرقم 75 من نظام Decimal إلى Hex وهو محور إهتمامنا

1- نقسم الرقم 75 على 16 والنتيجة يجب أن تكون رقماً صحيحاً، وعليه سيكون حاصل القسمة 4 والباقي 11

2- ومن خلال إجراء مقارنة بين الرقم 11 الذي يمثل الباقي وجدول المقارنة أعلاه لوجدنا أنه

يساوي الحرف B في نظام Hex
3-وعليه سيكون الرقم 75 في نظام decimal
يساوي B4 في نظام Hex.

نأخذ مثالاً آخر: تأمل الرقم مثلاً 127
نقسم الرقم 127 على 16 فنحصل على 7 كرقم صحيح والباقي 15. ومن خلال جدول المقارنة أعلاه نجد أن الباقي 15 يساوي F في Hex وعليه ستكون النتيجة النهائية:
127 في نظام Decimal يساوي F7 في نظام Hex.

نأخذ مثالاً آخر: مثلاً الرقم 500
نقسم أولاً على 16 فنحصل على 31 كرقم صحيح و4 كباقي، وبما أن ناتج القسمة والذي هو الرقم الصحيح 31 هو أكبر من 16، إذن نقسمه مرة أخرى على 16 فنحصل على 1 كرقم صيحح والباقي 15.
والآن لدينا باقيين، 4 من القسمة الأولى و15 من القسمة الثانية، الباقي الأول والذي هو 4 يبقى كما هو لأنه أقل من عشرة، بينما الباقي الآخر الذي هو 15 وبمقارنته مع جدول المقارنة أعلاه، نجد أنه يساوي الحرف F، لذلك ستكون النتيجة النهائية:
500 كرقم Decimal يساوي F41 في Hex
لاحظ أخي القارئ أن الباقي الأول والذي هو 4 جاء أول رقم من اليمين في النتيجة النهائية، أما الباقي الثاني والذي هو 15 أو الحرف F جاء ثانياً من اليمين بينما حاصل القسمة والذي هو الرقم 1 يأتي ثالثاً.

أرى أن أفضل وقت لعمل نسخة إحتياطية كاملة للريجستري بإستخدام خدمة تصدير التابعة لمحرر

الريجستري Regedit هي بعد أن يتم الإنتهاء من إنجاز كل الخطوات التالية وحسب تسلسلها:

1- ينصب ويندوز جديد على قرص نظيف وليس فوق نسخة قديمة من ويندوز.

2- تنصيب كافة مشغلات الأجهزة Hardware drivers ويفضل أن تكون الأحدث ما يكون

3- عمل التحديث للويندوز وللمتصفح إلى آخر تحديث إن كان ذلك ممكناً

4- تنصيب كل البرامج الضرورية مثل برامج الأمان والوقاية من الفايروسات وبرامج أوفيس ... إلخ

5- إستخدام برنامج خدمي لتصليح أخطاء الريجستري (من الضروري عمل إعادة تشغيل قبل وبعد إصلاح أخطاء الريجستري)

6- القيام بتنظيف القرص الصلب من الملفات الزائدة والميتة التي تأخذ حيزاً من القرص الصلب من دون لازم.

7- القيام بتصليح أخطاء القرص الصلب.

8- تسريع القرص الصلب أي عمل Defragmentation

9- تسريع عملية الإقلاع بإستخدام برنامج مايكروسوفت Bootvis

10- تصليح أخطاء الريجستري مرة ثانية

11- تشغيل محرر الريجستري ثم لا حظ الصورة التالية

لاستعادة النسخة الاحتياطيّة من الريجستري: إذهب إلى قائمة إبدأ Start، ثم إلى كل البرامج All Programs ثم إلى أدوات مساعدة Accessories ثم إلى أدوات النظام System Tools ومنها قم بإختيار إستعادة نظام System Restore. ومن أول نافذة تظهر قم بإنتقاء الخيار الأول Restore my computer to an earlier time ثم إضغط على Next، وعندها ستظهر لك النافذة الثانية وفيها أسماء نقاط الإرجاع وتأريخ إنشاءهم، ما عليك هنا إلا إختيار نقطة الإرجاع التي تريد إرجاع النظام إلى نقطتها ومن ثم تكملة المهمة إلى النهاية من خلال تتابع سير ظهور النوافذ وبالضغط إما على الزر Next أو OK.

****

*****

****

****

شرح مختصر لمحرر الريجستري Regedit

ذكرنا من خلال سياق الموضوع من أن البرنامج المساعد Regedit والذي يأتي مندمجاً مع الويندوز يستخدم لعمل تحرير أو إجراء تغيرات في الريجستري فقط. وقلنا كذلك بأن هذا البرنامج المساعد لا يمثل الريجستري نفسها على الإطلاق بل أن للريجستري ملفاتها الخاصة بها والتي يختلف عددها وأماكن تواجدها من ويندوز لآخر.

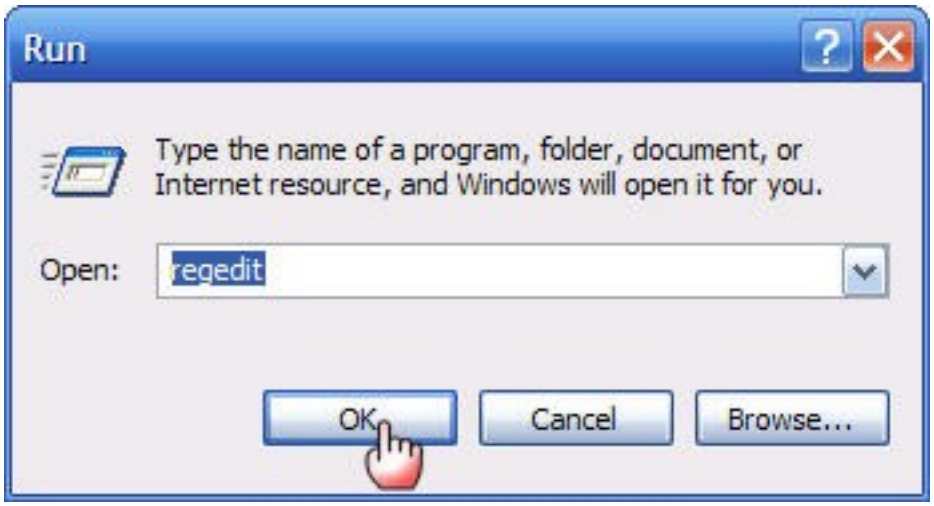
ولتشغيل هذا البرنامج المساعد إليك الطريقتين التاليتين:

الأولى: نستطيع عمل مختصر مسار Shortcut لهذا البرنامج المساعد على سطح المكتب، وبذلك نستطيع تشغيل هذا البرنامج في كل مرة نريد تشغيله فيها من خلال النقر مرتيين متتاليتين على أيقونته أي بنفس الطريقة التي نتعامل بها مع أي برنامج آخر. ولعمل مختصر مسار لهذا البرنامج المساعد نتبع الخطوات التالية الموضوحة بتسلسل الصور الثلاث الآتية:


Right click
على مكان فاضي
من سطح المكتب ثم
1
2
Arrange Icons By
Refresh
Paste
Paste Shortcut
Save as Scheme...
Graphics Options
Display Modes
New
Properties
Folder
Shortcut
Briefcase
Bitmap Image
Microsoft Word Document
WinRAR archive
Real-DRAW Project
Text Document
WAV Audio
WinRAR ZIP archive

الثانية: إذهب إلى الزر إبدأ Start، ثم إلى

تشغيل Run ثم أكتب Regedit ثم OK كما هو موضح في الصورة



وبغض النظر عن الطريقة المستخدمة لتشغيل محرر الريجستري فإن النتيجة ستكون واحدة، وهي ظهور الهرم التسلسلي للريجستري الذي قد يختلف من ويندوز لآخر ومن جهاز لآخر.

****

*****

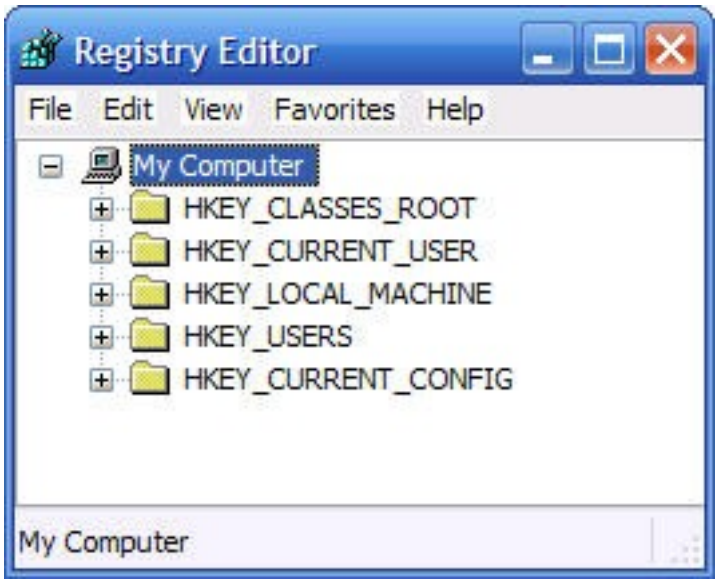
****

****

دعنا الآن نقوم بجولة سريعة في رحاب محرر الريجستري. لو أخذنا أولاً من قائمة ملف File الخياران إستيراد Import وتصدير Export. فهذان الخياران مفيدان جداً، فخيار التصدير يوفر طريقة ثانية في عمل نسخة إحتياطية للريجستري بكاملها أو لعمل نسخة إحتياطية لغصن كامل أو لعمل نسخة إحتياطية لفرع رئيسي أو حتى لمفتاح. وقد سبق لنا وأن تطرقنا إلى الطريقة الأولى لعمل نسخة إحتياطية من ريجستري الويندوز بإستخدام System Restore. وأما خيار إستيراد فيعمل عكس تصدير أي بإسترجاع أو دمج

أي جزء أو بمعنى آخر بإستيراد أي ملف ريجستري يحمل إمتداد .reg وفي الحقيقة أهمية الخيار تصدير هي الأهم بكثير من خيار إستيراد، لأن بمجرد ما يكون لديك ملف ريجستري فإنك سوف لن تحتاج لإستيراده عن طريق خيار إستيراد أي من دون الحاجة للذهاب إلى محرر الريجستري بل بإمكانك تشغيله مباشرة بنقرتي ماوس وهو في مكانه وينتهي الأمر. إذاً سيكون تركيزنا الآن على الخيار تصدير **Export**.

فلو أردنا تصدير أو عمل نسخة إحتياطية كاملة من الريجستري بإستخدام خيار تصدير ولمعظم إصدارات الويندوز نقوم بما يلي: نضغط بالماوس لمرة واحدة على **My Computer** أي لتضليلها فقط ثم نذهب إلى قائمة ملف ونختار تصدير **Export** .



****

******

*****

****

*******وبنفس الطريقة نستطيع تصدير أو عمل نسخة إحتياطية لغصن Hive كامل. فمثلاً نأخذ

الغصن الثاني HKEY_CURRENT_USER. نضغط بالماوس مرة واحدة على هذا الغصن لتضليله ثم نذهب إلى خيار تصدير من قائمة ملف ونكمل العملية بإعطاء إسم لملف الخزن مثلاً HKEY_CURRENT_USER من دون إمتداد لأن وكما قلنا أن عملية التصدير ستزفق الإمتداد reg لملف التصدير. ملف الريسجتري الناتج عن تصدير معلومات ريجستري هذا الغصن سيكون أقل أو بحدود من 10ميجابايت بالنسبة لويندوز إكسبي الحزمة الخدمية الثانية.

وبنفس الطريقة نستطيع تصدير معلومات ريجستري لأي فرع بالكامل هو وجميع الفروع والمفاتيح التي يحتويها أو التي يتكون منها. خذ مثلاً الفرع Keyboard Layout الموجود ضمن الغصن HKEY_CURRENT_USER.

وكما ترون فإن الفرع Keyboard Layout يتألف من أربعة فروع جانبية، وأحد هذه الفروع الجانبية والذي هو IMEtoggle يتفرع بدوره إلى فرع آخر scancode إنظر إلى الصورة. لعمل التصدير لجميع معلومات الفرع Keyboard Layout، نضغط مرة واحدة بالماوس على المجلد المسمى Keyboard Layout لتضليله، ثم نذهب إلى قائمة ملف ونختار تصدير Export ونسمي الملف مثلاً Keyboard Layout من دون إمتداد. حجم الملف الناتج عن عملية تصدير هذا الفرع سيكون صغيراً بالنظر لصغر الفرع نفسه وعدم تفرعه كثيراً.

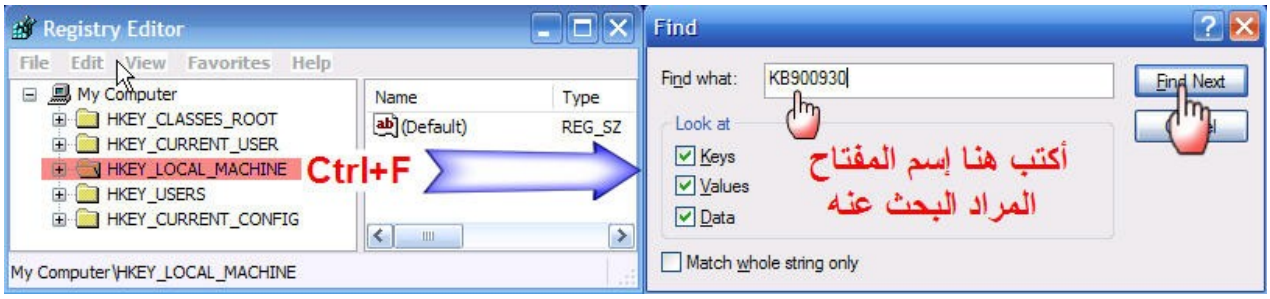
النص الأساسي وإلى الذين لديهم ويندوز إكسبي
النص المضاف

-/*) ولكن أعتمدت شركة مايكروسوفت محرر واحد لكل الأنظمة بداية من إصدارها windows 7 التالي لنظام Vista/-*) بإمكانهم تصدير أي جزء من الريجستري وذلك بعد تضليله، وبالضغط بيمين الماوس على مجلد ذلك الجزء ستظهر قائمة كاللتي في الصورة وعندها يمكن إنتقاء تصدير أو Export من هذه القائمة بدلاً من الذهاب إلى قائمة ملف وبعدها تستكمل عملية التصدير كالمعتاد بالطريقة التي تم شرحها أعلاه.

خيار البحث Find لمحرر الريجستري

في أغلب الأحيان يعتبر البحث عن إبرة في كومة قش مجهد وممل وقد لايصل إلى نتيجة، فكيف هو الحال إذا كان هناك عدت أبر متفرقات في كومة القش؟ هكذا في نظري يكون الواقع في الريجستري فيما لو قمنا بأنفسنا بالتفتيش عن مفتاح أو قيمة بين آلآف المفاتيح والقيم. ولله الحمد والمنة ثم لحسن الحظ يوفر محرر الريجستري لنا خيار البحث Find الآلي وخيار آخر ملحق به وهو خيار الإستمرار بالبحث Find Next. شغل محرر الريجستري بإحدى الطريقتين التي ذكرنا أعلاه، ومن الآن فصاعداً فسوف لن نكررعن كيفية تشغيل محرر الريسجتري على إعتبار أن طريقة التشغيل أصبحت مألوفة وبخاصة لمن يتابع الموضوع عن كثب. يتواجد خيار البحث الآلي لمحرر الريجستري في القائمة الرئيسية الثانية من الشمال (اليسار) أي في قائمة تحرير Edit.

وبإستخدام خيار البحث هذا الذي يوفره محرر الريجستري نستطيع القيام بنوعين من البحث الآلي: البحث العمومي والبحث الخصوصي. البحث الخصوصي يكون نطاق إستخدامه أقل نوعاً ما ويكاد يكون محصوراً على المحترفين أو العارفين بخبايا وأسرار الريجستري. فمثلاً، البعض يعلم بأن كافة المعلومات الخاصة بتسجيل تحديثات ويندوز إكسبي تكون متواجدة في الغصن الثالث أي HKEY_LOCAL_MACHINE. إذاً فليس هناك من داعي لتفتيش كل الريجستري للبحث عن هذا التحديث أو ذاك إختصاراً للوقت. فلو رغبت أن تعلم بأن التحديث KB900930 والذي يخص الحزمة الخدمية الثانية لإكسبي منصباً أو مسجلاً وبسرعة فما عليك إلا فعل التالي: وأنت في محرر الريجستري، إضغط مرة واحدة فقط على الغصن HKEY_LOCAL_MACHINE لتضليله، والآن لتشغيل ماكنة البحث لديك أمرآن إما الذهاب إلى القائمة الرئيسة تحرير Edit وإنتقاء بحث Find أو إستخدام الزرين Ctrl والحرف F من لوحة المفاتيح معاً وفي نفس الوقت ثم تابع الصورتين المتتاليتين من اليسار إلى اليمين



ومما تجدر الإشارة إليه أن أي تحديث يكون له عدت جذور هنا وهناك داخل هذا الغصن، لهذا يجب الإستمرار بالبحث الآلي من خلال إمكانية البحث الملحقة والتي هي Find Next أو الزر F3 من لوحة المفاتيح. أي يجب عدم الإكتفاء بنتيجة بحث واحدة بل الإستمرار في البحث إلى أن تحصل على رسالة مفادها لم يبقى من مفتاح يمكن إيجاده بعد ذلك اي تصل إمكانية البحث إلى النهاية. هذا فيما يخص البحث الخصوصي. أما فيما يخص البحث العمومي فأقول إنه يشبه تماماً البحث الخصوصي ولكنه أبطأ بكثير لكونه سيشمل البحث في الريجستري برمتها. والآن شغل محرر الريجستري، واضغط مرة واحدة على الساق My Computer لتضليله ثم شغل إمكانية البحث وأكتب إسم المفتاح المراد البحث عنه ثم أكمل البحث حتى النهاية عن طريق ملحق البحث. هذا النوع من البحث وبالرغم من أنه بطئ إلا أنه الأكثر شيوعاً بين العامة اللذين لديهم إلمام أساسي بالحاسوب.

****

*****

****

****

أساسيات ملف الريجستري

لملف الريجستري خصوصيات خارجية وداخلية. الخصوصية الخارجية الأولى تتضمن إنتهاء إسم الملف بالإمتداد reg، والذي على ضوءه يعطي الويندوز الأيقونة الخاصة والمعروفة بملف الريجستري لذلك الملف. وبكل سهولة الملف الذي ليس له إمتداد reg لا يعتبر ملف ريجستري بغض النظرعن أيقونته وبغض النظرعن المعلومات التي بداخله سواء أكانت صحيحة أم خاطئة. الخصوصية الخارجية الثانية هي الإسم قبل الإمتداد، والتي لها نفس شروط أسماء ملفات

الويندوز الأخرى وإن كان يفضل لملفات الريجستري أن تكون أسماءها قصيرة وواضحة ودالة على معنى ومن دون فراغات بين أحرف الإسم.
أما الخصوصيات الداخلية أو الشروط التي يجب أن تتوفر في تركيبة وترتيب المعلومات دخل الملف حتى يعتبر ملف ريجستري فيمكن التعرف عليها من خلال دراسة ملف الريجستري السهل التالي والخاص بويندوز إكسبي والأمثلة اللاحقة.
وظيفة هذا الملف هي التخلص من رسالة الخطأ التي غالباً ما تظهر عندما يصبح الحيز أو الفراغ المتبقي من القرص الصلب المنصب عليه ويندوز إكسبي أقل من 10% من الحجم الكلي للقرص.

الخصوصيات الداخلية

-1يجب أن يبدأ ملف الريجستري بالراس (Headوذلك ينطبق أيضاً على الويندوز 7 ومايليها) فهذا الرأس يعتبر بمثابة المحرك الذي يقوم بتشغيل محرر الريجستري والذي بدوره يقرأ المعلومات في الملف ومن ثم يطبقها أويضعها في مكانها المناسب في الريجستري. لكل ملف ريجستري رأس واحد مهما بلغ أو كبر ملف الريجستري ومهما كان عدد مفاتيحه. يفضل ترك سطر فارغ بين الرأس وما سيأتي بعده لزيادة الوضوح فقط.

-2بعد الرأس وبسطر أسفل منه يأتي المسار وهو الإتجاه الذي يسلكه محرر الريجستري في الريجستري للوصول إلى المكان المناسب لتطبيق المعلومات وإلا أين يذهب محرر الريجستري لكي يضع المعلومات. المسار بكامله يجب أن يبدأ وينتهي بقوس كبير [ ]، والسبب منطقي جداً هو حتى يعرف محرر الريجستري حدود المسار، أي أن فتح القوس يعني لمحرر الريجستري بداية المسار وسد أو غلق القوس يعنى لمحرر الريجستري نهاية المسار.

-3بعد المسار وبسطر أسفل منه مباشرة يأتي المفتاح أو الإسم ولكونه دالة إسم أو String يجب أن يكون داخل زوج من الفواصل أوالفارزات العليا أي "إسم المفتاح". وطالما أنه مفتاح يحمل قيمة إذاً يجب أن يكون هذا المفتاح من دون

فراغات أي كلمة واحدة مهما طالت.

-4بعد المفتاح وعلى نفس السطر ومن دون ترك أي فراغ على الإطلاق تأتي علامة اليساوي = والتي لاتعني كما في الرياضيات بأن كذا يعادل كذا بل تعني هنا كلمة جواب answer.

-5بعد اليساوي وعلى نفس السطر ومن دون ترك أي فراغ على الإطلاق يأتي منطوق الجواب، وله شقان:

الشق الأول: يتضمن التعريف بحجم ودقة لما سيكون عليه منطوق الجواب ويتمثل بالكلمة (dwordأنظر إلى موضوع أصل dword الموضح في الصورة أدناه).

القيمة العددية ل dword في النظام العشري العادي الذي نألفه هي 4294967296 أو 2 مرفوعة للقوة 32 (32 .(Bitأي ما معناه عندما تكون dword متواجدة قبل الرقم فإنها ستقوم بإبلاغ محرر الريجستري بأن الرقم الذي سيأتي بعدها هو رقم صحيح يتراوح بين الصفر كحد أدنى و 4294967296 كحد أقصى. بعد dword تأتي مباشرة : من دون أي فراغات.

الشق الثاني: نعود إلى مثالنا، وفيه سيكون منطوق الجواب بنعم Yes أو لا No. في عالم الكمبيوتر وفي الريجستري بالذات، غالباً يستعاض عن منطوق الجواب لا بالرقم صفر 0 والذي غالباً ما يكون الحالة الإفتراضية لكثير من المفاتيح أثناء تنصيب الويندوز، كما يستعاض عن منطوق الجواب نعم بالرقم واحد 1. وبما أن منطوق الجواب هنا في حالة مثالنا سيكون رقماً واحداً صغيراً متمثلاً بالرقم الواحد فيكفيه حجز مقعداً له بحجم 1 بايت (8 بت) من أصل 4 بايت ((dwordوتم تمثيله بالشكل 0000 0001. ولو كان منطوق الجواب يتطلب رقماً أكبر بكثير من الواحد لأمكن حجز 2 بايت (16بت)، أما عندما يكون منطوق الجواب يتطلب رقماً هائلاً فيجب حجز المقعد 4 بايت (32 بايت أو (dwordبكامله. قد يتساءل البعض لماذا لا يتم الحجز دائماً للأرقام ب 4 بايت (32 بت أو (Dwordلأنها صالحة لجميع الأرقام مهما صغرت أو كبرت، أقول هذا الكلام سليم ولكن حجز 4 بايت (32 بت) للأرقام الصغيرة كحالة المثال الذي بين أيدينا يعتبر إستهلاكاً للذاكرة في الوقت الذي من الممكن خفض إستهلاك الذاكرة إلى الربع بحجز 1 بايت (8 بت) الأكثر من الكافي.

في الحقيقة أنه بالإمكان كتابة قيمة المفتاح في مثالنا بطريقة ثانية كما في الصورة التالية

```
Windows Registry Editor Version 5.00

[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer]
"NoLowDiskSpaceChecks"="1"
```

وكما تلاحظ أخي القارئ أن قيمة المفتاح المتمثلة بالرقم واحد قد تم إحاطتها بفاصلتين عاليتين. والسبب في ذلك أن منطوق الجواب المتمثل بالرقم 1 خرج عن نطاق الأرقام وأصبح يُتعامل معه وكأنه حرف أو بالأحرى دالة إسم String أي أصبح بنفس وتيرة المفتاح الذي سبق وأن قلنا بأنه دالة إسم، لذلك وجب وضعه وسط فاصلتين عاليتين. قد يتسائل البعض أي الطريقتين أصح أو أدق أو أفضل: أقول كلتاهما صحيحتان ولكن الطريقة الأولى المتمثلة ب dowrd يفضلها المبرمجون والمحترفون لكونها دقيقة جداً وخالية من الأخطاء بينما الطريقة الثانية يعمل بها الكثير من المستخدمين العاديين للحاسوب لبساطتها وقد ينشأ عنها بعض الأخطاء ولكن بنسبة ضئيلة جداً

وكما قلنا سابقاً بأنه من الممكن صنع ملف ريجستري كبير يحوي على كل الإعدادات والخيارات المفضلة لدى مستخدم الكمبيوتر. فكما نلاحظ بأن الملف له رأس واحد وهذا هو المطلوب ولا يحتاج إلى وضع رأس ثان. للملف أيضاً عددت مسارات ولكن وكما هو واضح بأن هذه المسارات مرتبة بحسب تسلسل أولويات أغصانها Hives في محرر الريجستري. أي عندما نشغل محرر الريجستري فإن

الغصن HKEY_CURRENT_USER يأتي قبل الغصن HKEY_LOCAL_MACHINE ولذلك يجب أن ترتب المسارات في ملف الريجستري حسب تسلسل أوليات أغصانها. فالمسارات التابعة للغصن HKEY_CURRENT_USER يجب أن تأتي قبل المسارات التابعة الغصن HKEY_LOCAL_MACHINE وهكذا يكون ترتيب المسارات. أما فيما يخص ملاحظات التنبيه (الكلام باللون الأسود)، فقد حرصت على وضع ملاحظات قبل كل مسار بحيث يكون عدد الملاحظات بعدد المفاتيح، على ان توضح كل ملاحظة وحسب تسلسلها المفتاح الذي يناظرها بالتسلسل. المفاتيح الملونة باللون الأزرق لها منطوق جواب يتمثل برقم صحيح معرف ب dword وهذا سبق وأن ألقينا الضوء عليه بما فيه الكفاية. ونفس الشيء ينطبق على المفاتيح الملونة بالأخضر والتي لها منطوق جواب يكون إسماً أو حرفاً معرفة بدالة الإسم String.وما يميز هذان النوعان من المفاتيح هو أن المفتاح الذي يكون منطوق جوابه رقماً صحيحاً يجب أن يكون مسبوقاً بدالة التعريف dword بينما النوع الثاني لا يحتاج أن تكون الدالة String موجودة لأن الإسم والذي هو منطوق الجواب هنا سيكون محاطاً بفاصلتين عاليتين وهذا يكفل لمحرر الريجستري نسخ ما بين الفصلتين وتطبيقها بالكامل بغض النظر عن طبيعية الكلام الموجود. أنظر إلى الصورة التالية التي تبين كيفية ظهورهذين المفتاحين في محرر الريجستري وكيف أن أيقونتهما مختلفة. فالمفتاح الذي منطوق جوابه رقم صحيح والمعرف ب dword أيقونته فيها أرقام زرقاء للدلالة على أن منطوق الجواب يكون رقماً صحيحاً بينما المفتاح الأخر المعرف بدالة إسم String أيقونته فيها أحرف حمراء للدلالة على أن الإسم يكتب بالأحرف.

نأخذ مثالاً آخر نختتم به حصة أساسيات ملف الريجستري. في بعض الأحيان قد يتطلب الأمر رفع مفتاح معين من الريجستري بالكامل من جذوره من دون المساس بإمتداده ومن دون أي تأثير على أي مفتاح آخر في ملف الريجستري فكيف يتم ذلك؟ وما هو الأمر التنفيذي لذلك؟. لو قلت لكم لربما لا تصدقوني، الجواب هو فقط وضع علامة ناقص(-) بعد علامة اليساوي مباشرة من دون أي فراغ. خذ مثلاً لو أردنا توقيف التشغيل التلقائيبحيث لا تعمل هذه البرنامج مع كل إقلاع للويندوز وبذلك نوفر بضعت ميجابايت من الذاكرة بالإضافة إلى تسريع إقلاع الويندوز.

في أحيان أخرى قد يتطلب الأمر رفع مسار كامل من مكانه عندئذ سيتم رفع كل ما في ذلك المسار من مفاتيح وقيم. فمثلاً لو أردت رفع كل البرامج التي تشتغل تلقائياً مع إقلاع الويندوز ولم تكن تعلم بعدد وأسماء هذه البرامج إذن ما عليك إلا رفع خاصية تشغيل Run مع مسارها من الريجستري وهذا كفيل برفع وإيقاف أي برنامج ينطوي تحت لوائها من أن يشتغل تلقائياً مع إقلاع الويندوز. ولكن كيف؟ قد يبدو الأمر ولأول وهلة أنه صعب، ولكن لو قلت لكم كم هو سهل لما صدقتموني. والدليل تأمل الصورة التالية:

CLSID and ActiveX

في الأيام الأولى لولادة الإنترنت، كانت صفحات الإنترنت عبارة عن نصوص ورسوم ثنائية البعد بسيطة وغير متحركة وبفعل تقنية ActiveX، أصبحت المواقع وصفحات الإنترنت حية زاخرة بالعديدة من تقنيات الوسائط المتعددة المتقدمة Advanced Multimedia، والأجسام المتحركة التي يمكن التفاعل معها، والألعاب الثلاثية الأبعاد، ومشاهدة الفديو الحي بالصورة والصوت، بالإضافة إلى تشغيل العديد من البرامج المتقدمة ... إلخ. والهدف الأهم من هذا أن هذه التقنية ساهمت وإلى حد كبير في تطوير برامج إنترنت ومواقع لا تعتمد على نظام التشغيل المعتمد سواء أكان أبل ماكنتوش أو ويندوز أو لينكس أو غيره. أي بالإمكان التصفح والدخول إلى المواقع والتفاعل معها بغض النظر عن نوع نظام التشغيل المستخدم. وهذا بحد ذاته يعتبر إنجازاً رائعاً بكل المقاييس. فالكل يعلم أن برامج ماكنتوش للوسائط المتعددة والكرافيكس كانت ولفترة ليست بالقصيرة أفضل بكثير من برامج ويندوز، والآن قل الفارق، ولكن عندما تأتي الآن إلى عالم الإنترنت والتصفح بالذات فكلاهما يتساويان تقريباً.

ما هي تقنية ActiveX

قامت شركة مايكروسوفت في السنوات الأخيرة الماضية بتطوير وإضافة العديد من السمات على مشروعها القديم Win32 Application Programming Interfaces أو ما يعرف مختصراً APIs. ومن سمات التطوير التي تمت إضافتها هي دعم:

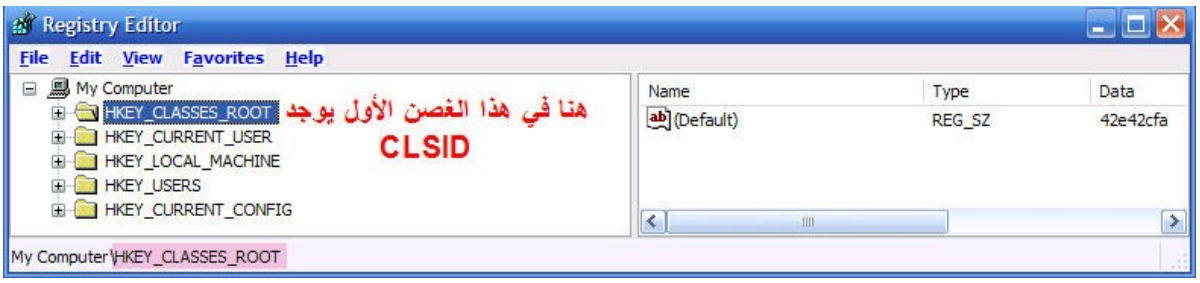
1- animation
2- 3D virtual reality
3- real-time audio
4- real-time video

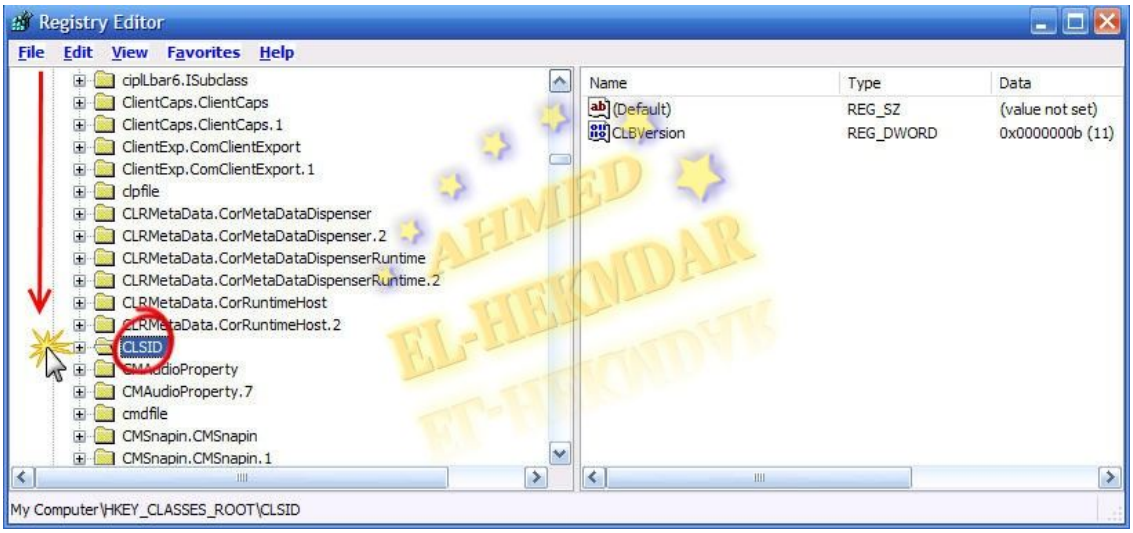
وأطلقت على الشكل النهائي للمشروع المطور إسم ActiveX. والغرض من هذا التطوير هو لدفع ومساعدة المطورين والمبرمجين من كتابة برامج إنترنت بحيث تدعم هذه البرامج الإضافات الأربعة المذكورة أعلاه. علماً بأن الإضافات الأربعة كانت محصورة ولفترة ليست بالقصيرة على البرامج العادية وليست على برامج الإنترنت. ولتقريب الفكرة إلى الأذهان، يمكن تشبيه تقنية ActiveX تماماً بتقنية مايكروسوفت Object Linking and Embedding أو مختصراً ب OLE. فتخيل مثلاً برنامج Word Microsoft الذي يدعم ميزة OLE وبرنامج آخر يعمل صور بحيثيدعم هو الآخر هذه الميزة أيضاً، إذن يمكن إدخال الصورة إلى برنامج Word وربطها بالصورة الأصلية في البرنامج الثاني بحيث أن أي تعديل يجرى على الصورة الأصلية ينعكس وفي نفس الوقت على الصورة المدخلة في Word. والآن لو قمت بإدخال عدة أجسام أخرى إلى برنامج Word مثل المعادلات الرياضية، والوسائط المتعددة وغيرها، سيصبح لديك في النهاية جسماً كبيراً صحيح أنه يمكن تشغيله ببرنامج word ولكن يحتوي على روابط لعدة برامج إشتركت في تكوينه. وأفضل طريقة لتمثيل وتسجيل هذا الجسم النهائي الحاوي على عدة روابط في ريجستري الويندوز هو بهيئة صف له عدد من الطلاب مساولعدد البرامج والملفات التي كونته بالإضافة إلى عنوان يميزه. وهذا ما يسمى بمميز

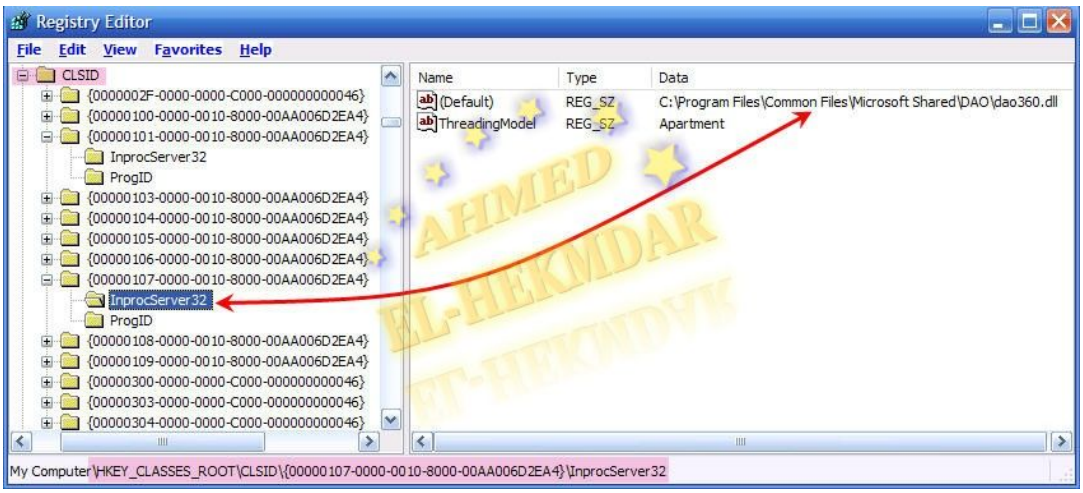
أو مشخص الصف Class Identifier أو مختصراً CLSID. ما معناه أن أي ملف أو برنامج تم تكوينه من أجسام تدعم ميزة OLE يجب أن يكون له مشخص صف في الريجستري. والحال ينطبق على تقنية ActiveX بالضبط لأن صفحات الإنترنت مكونة من نصوص وأجسام متحركة وأصوات وفيديو ... إلخ قامت بإنشاءها برامج مختلفة.

كل CLSID معرف في الريجستري بدالة الإسم String أو REG_SZ ويحجز مقعداً كبيراً مقداره 16بايت وليس 16بت. جميع CLSID في الريجستري لها نفس التركيبة، حيث يتكون كل CLSID من خمسة مجاميع من أرقام أو أحرف أو خليط من كليهما مكتوبة بنظام الأرقام هكس بحيث تفصلالمجموعات عن بعضها البعض بعلامة الناقص وتنحصر المجاميع الخمسة بين قوسين متوسطي الحجم أي { } . المجموعة الأولى تتألف من 8 عناصر، المجموعات الثانية والثالثة والرابعة يتألف كل منها من 4 عناصر بينما المجموعة الخامسة والأخيرة تتألف من 12 عنصراً.

يعتبر مشخص الصف أو CLSID من أعقد الفروع الرئيسية التابعة للغصن الأول HKEY_CLASSES_ROOT في الريجستري، كما أن هناك الكثير من CLSID في أماكن متفرقة من الريجستري. وقد يتفرع الفرع CLSID إلى فرع أو فروع جانبية بنفس الطريقة التي يتفرع فيها أي غصن أو فرع آخر.

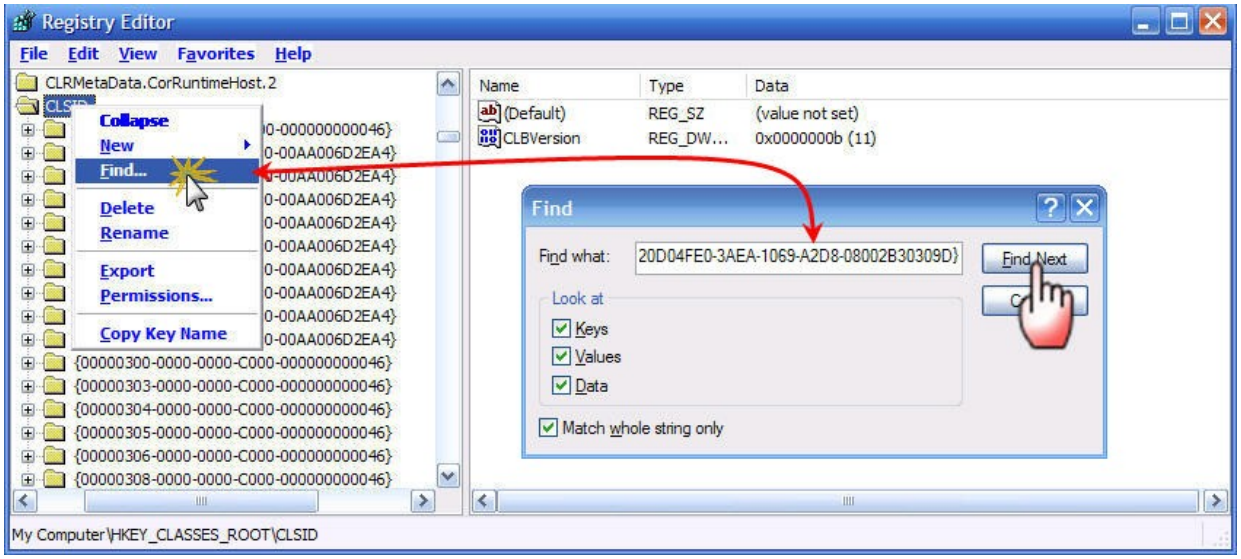


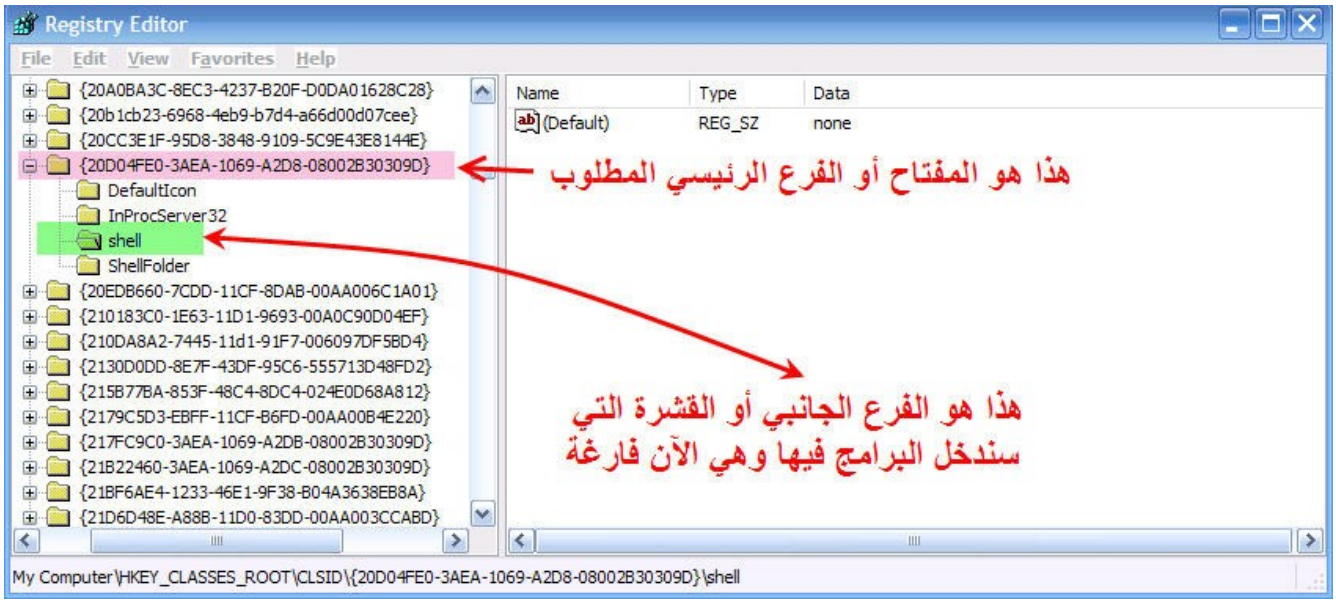




مثال: في هذا المثال على كيفية إضافة بعض البرامج المساعدة والخدمية إلى قائمة قشرة My Computer على سطح المكتب وذلك لتسهيل وتسريع التعامل مع هذه البرامج. في الحالات الطبيعية وإن كانت قد تختلف من جهاز لآخر، فإن الضغط بيمين الماوس على أيقونة My Computer المتواجدة على سطح المكتب فمن المفترض أن تظهر قائمة قد تبدو كالتالي:

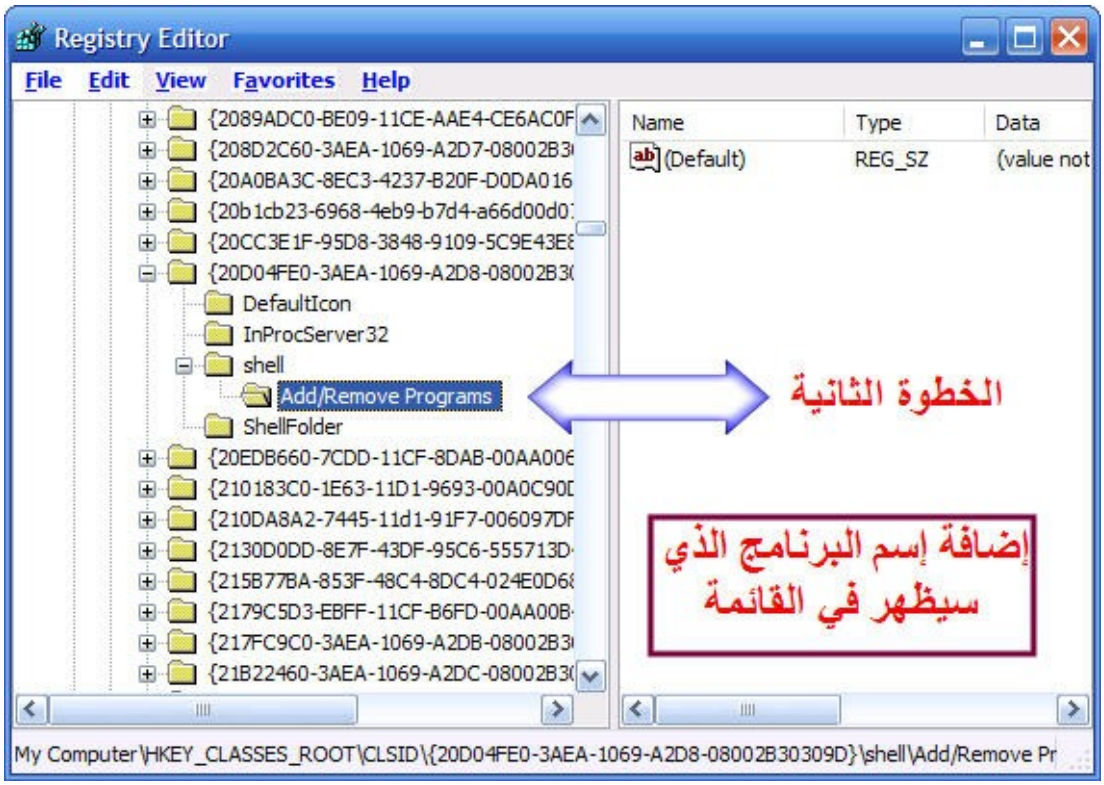
والآن إلى التطبيق في الريجستري حيث نقوم بتشغيل أولاً محرر الريجستري ثم نذهب إلى الغصن الأول HKEY_CLASSES_ROOT ثم نذهب إلى الفرع CLSID ثم نتوقف عنده. نضغط بعدها بيمين الماوس على أيقونة CLSID ونختار بحث أو Find من القائمة التي تظهر ثم نبحث عن المفتاح التالي:
{20D04FE0-3AEA-1069-A2D8-08002B30309D}

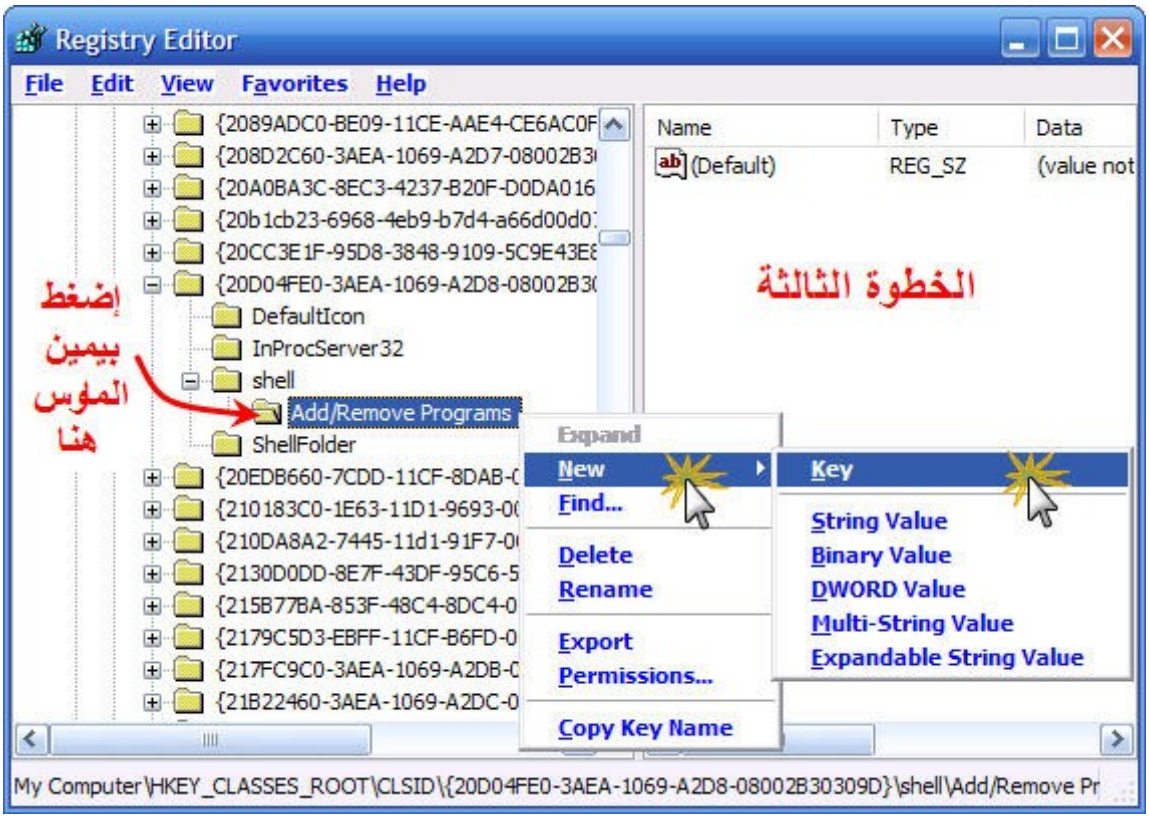




لكل برنامج نود إضافته إلى القشرة My Computer، يجب عمل مفتاحين، الأول يحمل إسم المفتاح الذي سيظهر في القائمة والمفتاح الثاني بداخل الأول ويتمثل بالأمر التنفيذي الذي سيقوم بتشغيل ذلك البرنامج. فلو أردنا مثلاً إضافة الخاصية Add/Remove programs نضغط بيمين الماوس على أيقونة القشرة Shell ثم نكمل بالصور


Registry Editor
File Edit View Favorites Help
{2089ADC0-BE09-11CE-AAE4-CE6AC0F
{208D2C60-3AEA-1069-A2D7-08002B3
{20A0BA3C-8EC3-4237-B20F-D0DA016
{20b1cb23-6968-4eb9-b7d4-a66d00d0
{20CC3E1F-95D8-3848-9109-5C9E43E
{20D04FE0-3AEA-1069-A2D8-08002B3
DefaultIcon
InProcServer32
shell
Sh
Expand
New
Find...
Delete
Rename
Export
Permissions...
Copy Key Name
Key
String Value
Binary Value
DWORD Value
Multi-String Value
Expandable String Value
Name
Type
Data
(Default)
REG_SZ
none
إضغط بيمين الماوس هنا
الخطوة الأولى
My Computer\HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell


Registry Editor
File Edit View Favorites Help
{2089ADC0-BE09-11CE-AAE4-CE6AC0F
{208D2C60-3AEA-1069-A2D7-08002B3
{20A0BA3C-8EC3-4237-B20F-D0DA016
{20b1cb23-6968-4eb9-b7d4-a66d00d0
{20CC3E1F-95D8-3848-9109-5C9E43E
{20D04FE0-3AEA-1069-A2D8-08002B3
DefaultIcon
InProcServer32
shell
Add/Remove Programs
ShellFolder
{20EDB660-7CDD-11CF-8DAB-00AA00E
{210183C0-1E63-11D1-9693-00A0C90
{210DA8A2-7445-11d1-91F7-006097DF
{2130D0DD-8E7F-43DF-95C6-555713D
{215B77BA-853F-48C4-8DC4-024E0D6
{2179C5D3-EBFF-11CF-B6FD-00AA00B
{217FC9C0-3AEA-1069-A2DB-08002B3
{21B22460-3AEA-1069-A2DC-08002B3
Name
Type
Data
(Default)
REG_SZ
(value not
الخطوة الثانية
إضافة إسم البرنامج الذي سيظهر في القائمة
My Computer\HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell\Add/Remove Pr


Registry Editor
File Edit View Favorites Help
{2089ADC0-BE09-11CE-AAE4-CE6AC0F
{208D2C60-3AEA-1069-A2D7-08002B3
{20A0BA3C-8EC3-4237-B20F-D0DA016
{20b1cb23-6968-4eb9-b7d4-a66d00d0
{20CC3E1F-95D8-3848-9109-5C9E43E
{20D04FE0-3AEA-1069-A2D8-08002B3
DefaultIcon
InProcServer32
shell
Add/Remove Programs
ShellFolder
{20EDB660-7CDD-11CF-8DAB-
{210183C0-1E63-11D1-9693-0
{210DA8A2-7445-11d1-91F7-0
{2130D0DD-8E7F-43DF-95C6-5
{215B77BA-853F-48C4-8DC4-0
{2179C5D3-EBFF-11CF-B6FD-0
{217FC9C0-3AEA-1069-A2DB-
{21B22460-3AEA-1069-A2DC-0
Name
Type
Data
(Default)
REG_SZ
(value not
الخطوة الثالثة
إضغط بيمين الماوس هنا
Expand
New
Find...
Delete
Rename
Export
Permissions...
Copy Key Name
Key
String Value
Binary Value
DWORD Value
Multi-String Value
Expandable String Value
My Computer\HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell\Add/Remove Pr


Registry Editor
File Edit View Favorites Help
{2089ADC0-BE09-11CE-AAE4-CE6AC0F
{208D2C60-3AEA-1069-A2D7-08002B3
{20A0BA3C-8EC3-4237-B20F-D0DA016
{20b1cb23-6968-4eb9-b7d4-a66d00d0
{20CC3E1F-95D8-3848-9109-5C9E43E8
{20D04FE0-3AEA-1069-A2D8-08002B3
DefaultIcon
InProcServer32
shell
Add/Remove Programs
command
ShellFolder
{20EDB660-7CDD-11CF-8DAB-00AA00E
{210183C0-1E63-11D1-9693-00A0C90
{210DA8A2-7445-11d1-91F7-006097DF
{2130D0DD-8E7F-43DF-95C6-555713D
{215B77BA-853F-48C4-8DC4-024E0D6
{2179C5D3-EBFF-11CF-B6FD-00AA00B
{217FC9C0-3AEA-1069-A2DB-08002B3
مفتاح الأمر الذي قمنا بإنشاءه
Name
Type
Data
(Default)
REG_SZ
(value not set)
ننقر مرتين بالماوس هنا
الخطوة الرابعة
My Computer\HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell\Add/Remove Programs\command


Edit String
الخطوة الخامسة
Value name:
لاحظ لا يوجد إسم هنا
Value data:
control appwiz.cpl
نكتب الأمر هنا
OK
Cancel


Registry Editor
File Edit View Favorites Help
{2089ADC0-BE09-11CE-AAE4-CE6AC0F
{208D2C60-3AEA-1069-A2D7-08002B3
{20A0BA3C-8EC3-4237-B20F-D0DA016
{20b1cb23-6968-4eb9-b7d4-a66d00d0
{20CC3E1F-95D8-3848-9109-5C9E43E
{20D04FE0-3AEA-1069-A2D8-08002B3
DefaultIcon
InProcServer32
shell
Add/Remove Programs
command
ShellFolder
{20EDB660-7CDD-11CF-8DAB-00AA00E
{210183C0-1E63-11D1-9693-00A0C90
{210DA8A2-7445-11d1-91F7-006097DF
{2130D0DD-8E7F-43DF-95C6-555713D
{215B77BA-853F-48C4-8DC4-024E0D6
{2179C5D3-EBFF-11CF-B6FD-00AA00B
{217FC9C0-3AEA-1069-A2DB-08002B3
Name
Type
Data
(Default)
REG_SZ
control appwiz.cpl
لاحظ أن العنوان
الإفتراضي بقى
لاحظ الأمر التنفيذي له إمتداد
cpl (Control Panel)
تعريف دالة الإسم لأن القيمة عبارة عن إسم ثابت
My Computer\HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell\Add/Remove Programs\command

مثال آخر كإعادة لطريقة المثال الأول: لو أردنا إضافة محرر الريجستري إلى قشرة Computer، نعود مرة ثانية إلى القشرة Shell ونضغط بيمين الماوس على أيقونتها ثم My نكمل كما في الصور

والآن نقوم بإنشاء مفتاح آخر داخل المفتاح الأول الذي قمنا بإنشاءه قبل قليل ونضع له عنواناً. هذا العنوان يكون ثابتاً دائماً مهما إختلف إسم البرنامج المراد إضافته والعنوان الثابت هو الأمر، أما القيمة التي سيأخذ الأمر command فهي التي ستختلف من برنامج لآخر لكونها تمثل أمر تشغيل ذلك البرنامج

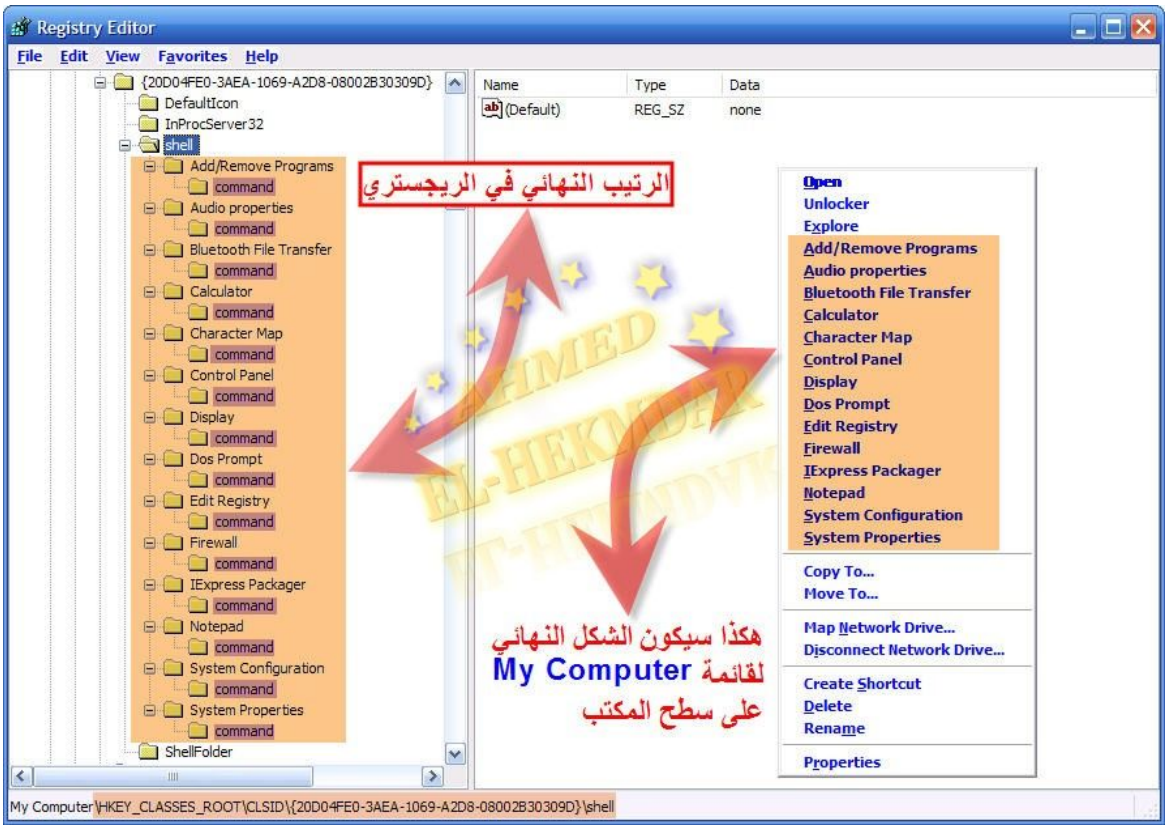

Registry Editor
File Edit View Favorites Help
Name Type Data
(Default) REG_SZ (value not set)
القيمة الإفتراضية لا زالت فارغة لحد هذه اللحظة
لاحظ أننا سنقوم بإنشاء مفتاح الأمر داخل المفتاح
الذي قمنا بإنشاءه في الخطوة السابقة وسنعطي
للمفتاح الجديد العنوان command
ويجب أن نعطي هذه المرة قيمة لهذا المفتاح
والتي تتمثل بالأمر التنفيذي لتشغيل ذلك البرنامج
نضغط بيمين الماوس هنا
DefaultIcon
InProcServer32
shell
Add/Remove Programs
Edit Registry
ShellFolder
Expand
New
Find...
Delete
Rename
Export
Permissions...
Copy Key Name
Key
String Value
Binary Value
DWORD Value
Multi-String Value
Expandable String Value
My Computer\HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell\Edit Registry


Registry Editor
File Edit View Favorites Help
Name Type Data
(Default) REG_SZ (value not set)
في هذه اللحظة مفتاح الأمر ليس له قيمة،
ولإعطاءه قيمة تتمثل بأمر التنفيذ ننقر مرتين
بالماوس على أيقونته
لاحظ أننا الآن إنتهينا من إنشاء مفتاح
الأمر وأعطيناه العنوان command
ولكن بقي علينا هذه المرة أن نعطي قيمة
لمفتاح الأمر حتى يقوم بتشغل البرنامج
DefaultIcon
InProcServer32
shell
Add/Remove Programs
Edit Registry
command
ShellFolder
My Computer\HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell\Edit Registry\command

وعند الضغط بيمين الماوس على أيقونة My Computer المتواجدة على سطح المكتب نحصل على الشكل التالي

وبإمكان المتابع وبنفس تسلسل الطريقة أعلاه إضافة البرامج والأوامر التنفيذية التالية إن رغب بذلك وإن أحب أن يزيد عليها فهذا أمر طيب يعود إليه:

1- إضافة الحاسبة Calculator وأمرها التنفيذي هو calc
2- إضافة لوحة التحكم Control Panel وأمرها التنفيذي هو control
3- إضافة Dos Prompt وأمرها التنفيذي cmd.exe
4- إضافة System Properties وأمرها التنفيذي control sysdm.cpl
5- إضافة Display وأمرها التنفيذي control desk.cpl
6- إضافة firewall وأمرها التنفيذي control firewall.cpl
7- إضافة Audio properties وأمرها التنفيذي control mmsys.cpl
8- إضافة Character Map وأمرها التنفيذي charmap.exe
9- إضافة Notepad وأمرها التنفيذي notepad.exe
10- إضافة IExpress Packager وأمرها التنفيذي iexpress.exe
11- إضافة System Configuration وأمرها التنفيذي msconfig.exe
12- إضافة Bluetooth File Transfer وأمرها التنفيذي fsquirt.exe



ول CLSID فوائد وتطبيقات أخرى كثيرة جداً لا يسع المجال للتطرق إليها، منها في تسريع وتصحيح أخطاء المتصفح والطباعة، ومنها أيضاً يعمل على حل مشاكل الويندوز. وإن شاء الله إذا سنحت الفرصة في المستقبل لعمل كل هذه المسائل بهيئة حلول ثابتة فلن نقصر بذلك بعونه تعالى.

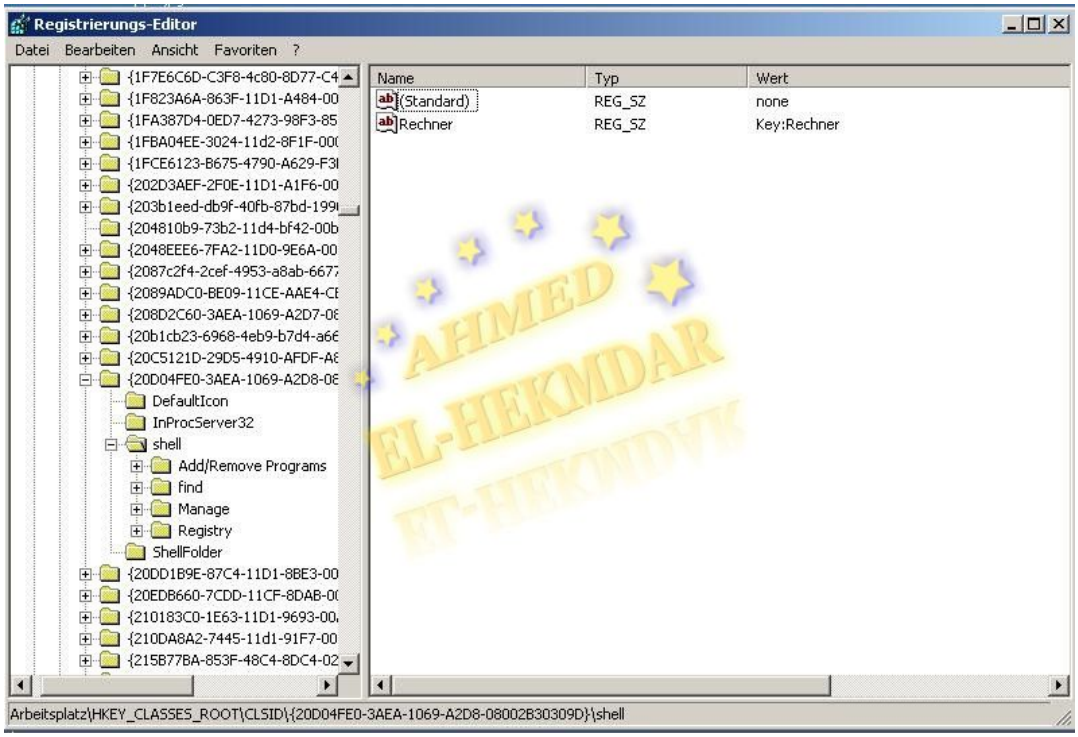
وقبل مغادرة بند CLSID لابد من التطرق وبإختصار إلى أن أهمية هذا الفرع قد إزدادت في السنوات الأخيرة حيث لا حظت الشركات أن أفضل وسيلة لتسجيل إنتاجاتها من البرامج وبخاصة برامج الوسائط المتعددة Multimedia والكرافيكس وغيرها في الريجستري هي بإستخدام CLSID. وفعلاً أن هذه الطريقة قد وضعت حد للناس العاديين من إيجاد الدواء اللازم للبرامج. لأن المستخدم العادي للحاسوب يذهب عادة إلى الريجستري للبحث عن البرنامج الذي نصبه لإيجاد الدواء الشافي له، وغالباً ما يستخدم إسم البرامج أو إسم الشركة المنتجة للبرنامج للبحث في الريجستري، ولكن كيف إذا تم إستبدال إسم البرنامج ومعلومات تسجيله ب CLSID الذي يحتوي على خمسة مجاميع مختلطة من أرقام هكس! بالنسبة للمحترفين فلن تشكل تلك الخطوة المعتمدة من قبل الشركات

المنتجة للبرامج عائقاً يذكر في طريقهم نحو إيجاد الدواء الشافي (لكل داء دواء).
وقبل تنصيب أي برنامج يطمحون في إيجاد الدواء له، يعمل المحترفون على مراقبة الريجستري ببرنامج مثل مراقب الريجستري Registry monitor أو Regmon وكثير غيرهم، حيث يقوم برنامج مراقبة الريجستري أولاً بعمل نسخة إحتياطية للريجستري، ثم يبدأ برقابة صارمة على الريجستري مسجلاً كل حدث يجري في الريجستري مهما كان صغيراً، وبالإمكان عرض حالة الريجستري الآنية وما يحدث فيها من تغير على الشاشة في كل لحظة. وبعد تنصيب البرنامج المطلوب إيجاد العلاج الشافي له، يصنع نسخة إحتياطية جديدة للريجستري. وبإجراء عملية الطرح للنسخة القديمة من النسخة الجديدة نحصل على الفارق الذي يمثل معلومات تسجيل البرنامج في الريجستري. وبعد ذلك تجرى بعض التعديلات على معلومات التسجيل لإيجاد الدواء الشافي للبرنامج والتي غالباً ما تكون سهلة. أرجو أن أكون قد أجبت وبإختصار على تساؤلات بعض الإخوة في هذا الموضوع، وفي نفس الوقت أعتذر عن تقديم أي تفصيلات آخرى في هذا الصدد.
وبهذا نكون وبحمد من الله سبحانه وتعالى قد إنتهينا من إلقاء الضوء على موضوع CLSID، وقد يكون لنا عودة إليه إن سنحت الفرصة لذلك في المستقبل القريب.
حصتنا القادمة ستكون إن شاء الله تعالى عن دور الريجستري في التنصيب التلقائي لويندوز إكسبي والبرامج التلقائية التنصيب.

السؤال :
حاولت أن أضيف key جديد في المسار
[HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell]
بدون الدخول إلى محرر الرجستري أي أنني فتحت ملف نص جديد وكتبت فيه التالي:

```
Windows Registry Editor Version 5.00
;Rechner
[HKEY_CLASSES_ROOT\CLSID\{20D04FE0-3AEA-1069-A2D8-08002B30309D}\shell]
"Rechner"="Key:Rechner"
```

ثم ذهبت إلى نفس المسار ووجدت أنه قام بصنع الملف الموجود في الصورة



فكيف لي أن أنشئ key بدون الدخول إلى محرر الرجستري؟؟
ملاحظة: كلمة Rechner تعني الحاسبة

تخيل أنك تريد عمل مجلد إسمه Shell وليس فيه ملف، ثم تقوم بإنشاء مجلد آخر في المجلد Shell وتسمي المجلد الجديد ب Rechner، وبداخل المجلد Rechner وضعت ملف الحاسبة أو برنامج الحاسبة. إذن هناك مجلدين الواحد بداخل الآخر وفي داخل الأخير ملف تشغيل. طالما أن المجلد الأول Shell ليس فيه ملفاً تنفيذياً إذن يجب أن نضع علامة @ بعدها يساوي ثم الكلمة لا شئ أو none طبعاً بين فواصل عالية. أما المجلد الثاني Rechner والذي يحمل إسم الحاسبة في القائمة فإنه يحتوي على ملف تشغيل الحاسبة ولهذا يجب أن نضع علامة @ بعدها يساوي ثم يأتي إسم البرنامج التشغيلي الذي هو Calc وبالطبع بداخل فواصل عالية.

****

***** **********

****

علاقة الريجستري بالتنصيب التلقائي لويندوز إكسبي والبرامج والتحديثات

في الأعوام القليلة الماضية برز وبشكل ملحوظ التنافس على عمل أفضل قرص سيدي لتنصيب ويندوز إكسبي تلقائياً ومعه عدد من البرامج والتحديثات التلقائية التنصيب أيضاً وبأقل خطأ ممكن. ومما تجدر الإشارة إليه أننا سوف لن نتكلم هنا عن عملية صنع السيدي التلقائي التنفيذ فهذا موضوع آخر، ما سنتطرق إليه في هذه الحصة هو دور الريجستري المهم في عملية صنع السيدي التلقائي التنفيذ فقط. وقد نخصص دروساً آخرى لعمل أفضل سيدي تلقائي التنفيذ في المستقبل القريب إن شاء الله تعالى.عندما يبدأ تنصيب ويندوز إكسبي، وفي طوره الأول المتمثل بنسخ الملفات من قرص السيدي إلى مجلد الويندوز في القرص الصلب، يتم نسخ ملفات الريجستري أيضاً التي لا تكون فارغة بل فيها أصلاً بعض المعلومات التي تخص تسجيل الملفات الأساسية المكونة للنظام والتي لا يمكن إستبدالها أو رفعها فيما بعد. ملفات الريجستري المنسوخة من قرص السيدي إلى مجلد النظام في القرص الصلب تكون في بادئ الأمر مضغوطة بنسبة أكثر من 70%، وبإنتهاء الطور الأول والذي هو طور النسخ وإعادة التشغيل وبدأ الطور الثاني، فأول ما يقوم به محرك تنصيب الويندوز هو فك الضغط عن ملفات الريجستري المضغوطة وغيرها من الملفات. وبعدها يبدأ محرك تنصيب الويندوز بتنصيب مشغلات الأجهزة Hardware drivers وتسجيل معلوماتها في الريجستري. بعد ذلك يبدأ محرك تنصيب الويندوز بتنصيب البرامج المساعدة ومتصفح الويندوز Windows Explorer ثم

متصفح إنترنت إكسبلور Internet Explorer ثم Windows Media Player إلى أن يتم تنصيب وتسجيل جميع الملفات والبرامج الخاصة بالويندوز. وبإنتهاء هذه المرحلة أي مرحلة تنصيب البرامج والملفات وتسجيلها يكون الويندوز قد وصل إلى العشرة دقائق الأخيرة المتبقية من عملية التنصيب وعندها تدخل عملية تنصيب الويندوز المرحلة الأخيرة وهي طور تسجيل المكونات وفحص وإختبار معلومات الريجستري والتأكد من صلاحية النظام وغيرها. إذا متى يجب أن تبدأ عملية تنصيب البرامج الإضافية وتحديثات الويندوز المراد تنصيبها تلقائياً وما هو دور الريجستري في ذلك؟

منطقياً بل ضرورياً أن تبدأ عملية تنصيب البرامج الإضافية وتحديثات الويندوز المراد تنصيبها تلقائياً تماماً قبل بدأ المرحلة الأخيرة من تنصيب الويندوز. وهذا ما يحدث فعلاً، فقد خصصت مايكروسوفت الثلاث دقائق ما قبل بدأ المرحلة الأخيرة أي عند حوالي الدقيقة 13 من النهاية لبدأ عملية قدح محرك تنصيب البرامج الإضافية والتحديثات المراد تنصيبها تلقائياً. ولكن لماذا في هذا الوقت بالذات:

-1لأن الويندوز والريجستري قد إكتملا في هذا الوقت، والريجستري بالذات جاهزة لتلقي التعديلات والإضافات إليها.

-2عملياً، لا يمكن تنصيب تحديثات الويندوز وإنترنت إكسبلور في وقت مبكر آخر لانهما غير مكتملين بعد.

-3بعض البرامج والأدوات المساعدة تحتاج بعضها للبعض الآخر، مثلاً تنصيب بعض البرامج وبخاصة التي لها إمتداد msi يحتاج أن يكون Windows Installer منصباً وجاهزاً للإستخدام، ولا يكون ذلك ممكناً إلا في هذه المرحلة بالذات

دور الريجستري

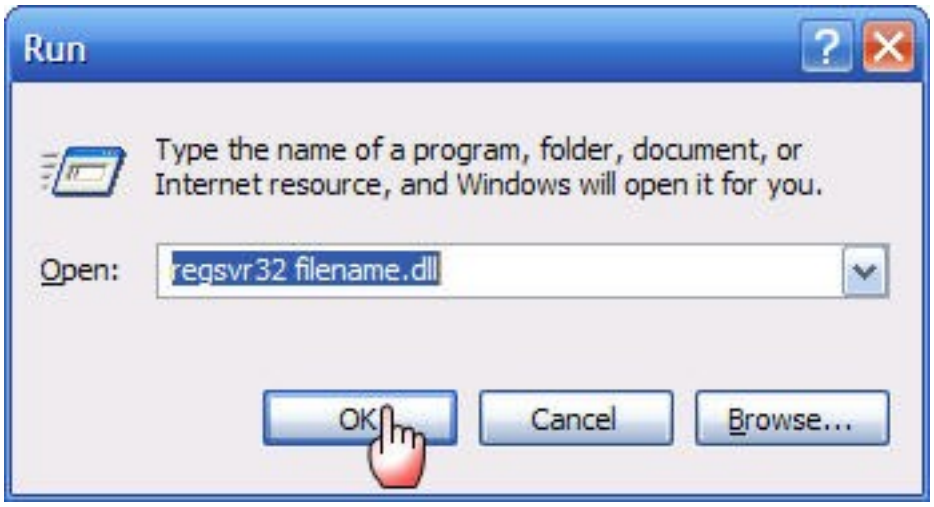
يمكن تقسيم دور الريجستري في التنصيب التلقائي للويندوز والتحديثات والبرامج إلى شطرين
الشطر الأول: وهو ما سنتكلم عنه في هذه الحصة ويعتمد على إستخدام ملفات الريجستري الجاهزة المعدة سابقاً إما:
-1لتسجيل معلومات برنامج معين في الريجستري، مثلاً لو كان لديك الرقم السري وكلمة عبور مع مساراتها في الريجستري لبرنامج معين فيمكن أن تضع هذه المعلومات في ملف ريجستري بحيث يتم تشغيله تلقائياً في أي لحظة بعد تنصيب ذلك البرنامج، وبذلك يتم تسجيل البرنامج تلقائياً من دون الحاجة لإدخال معلومات التسجيل فيما بعد وباليد.
-2لتطبيق تعديلات وإعدادت تفضيلية للويندوز تم جمعها وإعدادها بالشكل الذي يناسب ذوق ومتطلبات عمل مستخدم الحاسوب. مثلاً كأن تقوم بجمع كل إعدادات الريجستري المفضلة لديك من ويندوز مريض قبل مسحه ووضعها بملف ريجستري واحد لكي يتم تطبيقها على الويندوز الجديد تلقائياً وبذلك يحتفظ الويندوز الجديد بالإعدادت المفضلة لديك.والآن إلى الأوامر
الأمر التنفيذي الذي يستخدم عادة لقدح وتشغيل ملف الريجستري أثناء التنصيب التلقائي للبرامج يكون كالآتي:

فمثلاً لو كان لديك ملفي ريجستري، الأول يحتوي على ملعومات تسجيل فلاش جيت وإسمه FlashGet.reg والملف الثاني يحتوي على إعدادات تفضيلية لتحسين أداء ويندوز إكسبي وإسمه XPbest.reg وكلا الملفين موضوعين في مجلد عنوانه Mybest في القرص الصلب الذي سيحوي على مجلد الويندوز، كيف سيتم قدح وتشغيل هذين الملفين أثناء التنصيب التلقائي، أي بمعنى ما هي الأوامر التي بموجبها سيتم تشغيل هذين الملفين؟

هناك أمر تنفيذي آخر للريجستري يخص التنصيب التلقائي للبرامج وهو في غاية الأهمية ولكن للأسف يجهله الكثيرون. هذا الأمر التنفيذي يدعى regsvr32 ووظيفته تسجيل ملفات dynamic-link libraries أو dll في الريجستري. فمثلاً لو كان لديك ملف dll قمت بتحميله بصورة إنفرادية من هنا أو هناك وكانت هذه النسخة أحدث من النسخة المنصبة في جهازك أو أن التي في جهازك تالفة ويتطلب إستبدالها، فكل ما عليك هو نقل النسخة الجديدة إلى مجلد system32 في مجلد ويندوز ثم تسجيل الملف الجديد في الريجستري عن طريق هذا الأمر التنفيذي. ومما تجدر الإشارة إليه إلى أن هذا الأمر التنفيذي للريجستري يمكن إستخدامه في أي وقت وليس فقط في التنصيب التلقائي للويندوز والتحديثات والبرامج. والشكل العام الذي يجب أن يكون عليه هذا الأمر التنفيذي للريجستري أثناء التنصيب التلقائي للويندوز

أما الشكل الذي يمكن أن يكون عليه هذا الأمر التنفيذي للريجستري عند تشغيله في الويندوز فيكون بالشكل

التالي: إذهب إلى قائمة إبدأ Start، ثم تشغيل Run ثم نكمل كما في الصورة



لاحظ عزيزي أنه في كلتي الحالتين لم نقم أو لم نشر إلى مسار ملف dll، لان على الأغلب يكون مكان تواجد هذه الملفات هو المجلد system32، وكما ذكرنا سابقاً أن أي ملف في هذا المجلد يمكن إستدعائه مباشرة من دون الحاجة إلى ذكر مساره. ومن يريد وضع المسار فليضعه فهذا ليس خطأ ولكنه زائداً. أما إذا كان ملف dll موجوداً خارج مجلد system32 فيجب ذكر المسار بالكامل وإلا لن يتم التسجيل المطلوب.

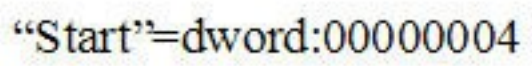
وفي هذا الشطر سنقوم إن شاء الله تعالى وبدلاً من أن نستخدم ملف ريجستري جاهز سنقوم بكتابة الأومر التنفيذية التي نريد تطبيقها على الريجستري مباشرة.

1-إضافة مفتاح جديد أو إجراء تعديل على قيمة مفتاح موجود أصلاً في الريجستري: مثال دعنا نقوم بوقف النوافذ الدعائية المزعجة المرافقة للمسنجر، علماً بأن

إيقاف النوافذ المزعجة لا يؤثر على الإطلاق على عمل الماسنجر نفسه بل بالعكس يحسن أداءه. مسار المفتاح في الريجستري والذي بموجبه يمكن وقف النوافذ المزعجة هو

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Servic es\Messenger

أما عنوان المفتاح وقيمته التي توقف النوافذ المزعجة



والآن ما هو الأمر التنفيذي المباشر والوحيد والتلقائي الذي سيقوم بالذهاب إلى الريجستري ويقوم بصنع هذا المفتاح إن لم يكن موجوداً أو بتعديل قيمته إن كان موجوداً أصلاً من دون أي إشعار:

reg add HKLM\CurrentControlSet\Services\Messenger /v Start /t REG_DWORD /d 4 /f

وإلى التفاصيل من الشمال (اليسار) إلى اليمين:
يبدأ الأمر بكلمة reg وهي إختصار ل Regedit ومن خلاله يتم إستدعاء محرر الريجستري لتنفيذ ما سيأتي بعدها. يعقب ذلك فراغ واحد ثم نوع التنفيذ الذي سيقوم به محرر الريجستري، وهنا إضافة add، أي سواء كنت ترغب بإضافة مفتاح جديد أو إجراء تعديل على قيمة مفتاح موجود أصلاً عليك إستخدام الأمر add.يأتي بعد ذلك فراغ واحد ثم ذكر مسار المفتاح في الريجستري بالكامل.
والمسار في حالتنا هو

HKLM\CurrentControlSet\Services\Messenger

لاحظ أننا إستخدمنا هنا المختصر HKLM بدلاً عن HKEY_LOCAL_MACHINE وهذا جائز هنا مثل إستخدامنا للمختصر reg بدلاً عن Regedit. بعد المسار في الريجستري يأتي فراغ واحد قبل وبعد v/، حيث يقوم الأمر v/ بإبلاغ محرر الريجستري إما بعمل المفتاح إن كان غير موجود أو بتعديل قيمته إن كان موجوداً أصلاً. بعد ذلك يأتي عنوان المفتاح والذي في حالتنا Start، يأتي بعد ذلك فراغ واحد ثم يأتي t/ ثم فراغ واحد بعدها أيضاً. فائدة t/ هو إبلاغ محرر الريجستري بنوع تعريف منطوق الإجابة للمفتاح، أي هل أن الجواب سيكون رقماً صحيحاً بنظام الأرقام هكس معرفاً بالدقة REG_DWORD أم إسماً معرفاً بدالة الإسم String، وبما أن الجواب أو قيمة المفتاح في حالتنا هذه سيكون رقماً صحيحاً إذن يجب أن نستخدم تعريف الدقة REG_DWORD. بعد التعريف يأتي فراغ واحد ثم الأمر d/ ثم فراغ واحد أيضاً. وفائدة الأمر d/ أو بالعربي الأمر إفعل ما تأمر به أي do بعد ذلك تأتي قيمة المفتاح والتي هي في حالتنا 4 لكي تقوم بوقف النوافذ المزعجة المرافقة للمسنجر، وأخيراً وبعد قيمة المفتاح يأتي فراغ واحد ثم الأمر f/ من دون أن يعقبة أي فراغ بعده، وفائدة الأمر f/ هو جعل تنفيذ الأمر برمته تلقائياً صامتاً أو هادئاً بمعنى آخر يقوم هذا الأمر بلجم أي إشارة تنبيه قد تنشأ عن تنفيذ هذا الأمر أثناء التنصيب التلقائي.

2- إزلة أو رفع مفتاح معين غير مرغوب فيه من الريجستري. هنا سنتطرق إلى مثال يقوم الأمر التنفيذي فيه بإزالة ما بات يعرف ب Microsoft Spyware

المندمج أصلاً بمتصفح إكسبلور التابع لويندوز إكسبي الحزمة الخدمية الثانية. قد يستغرب البعض من ذلك ولكن هذه حقيقة باتت مكشوفة للكثيرين. وبإمكانك التأكد من ذلك بعد تنصيب ويندوز إكسبي الحزمة الخدمية الثانية مباشرة أن تقوم بإستخدام أي برنامج لإزالة التجسس، فسوف ترى أنه يظهر خيار رفع ملف التجسس التالي .Alexa Spywareعمل ملف التجسس هذا معقد للغاية، فهو لا يعمل في جميع الأوقات، فبعد أن تزورعدد من المواقع وقام جهازك بخزن عناوين الموقع التي قمت بزيارتها، ثم قمت بزيارة أحد المواقع التي قد تستغرق بعض الوقت لإظهار صفحتها، يقوم ملف التجسس هذا بإنتهاز الفرصة أثناء وقت قراءة صفحة الموقع الطويلة بإرسال معلومات مهمة عن المواقع التي زرتها وعن جهازك إلى MSN ومنها إلى Alexa الشركة المنتجة لهذا النوع من ملفات التجسس لغرض إستخدامها في قضايا دعائية وأخرى لا يعلمها إلا الله سبحانه وتعالى ومن ثم اللذين قاموا ببرمجته. لاحظ عزيزي القارئ الحلقة الغامضة في التجسس، فلا أحد يعرف أين تنتهي هذه الحلقة، من مكان إلى آخر والنهاية مفتوحة، فالأمر لا ينتهي عند Alexa بل أنا متأكد من أن هناك العديد من الأماكن المستفيدة من هذا التجسس. وهذا هو الوجه الإحترافي في التجسس حيث أنك لا تستطيع مسك رأس الأفعى، فهل أن مايكروسوفت وضعته متعمداً وهي المستفيد الأول والأخير منه بصورة غير مباشرة عن طريق الوسطاء، أم هي لا تعلم بأثره السلبي!!!!!!!. على أية حال نعود بكم الآن إلى الأمر التنفيذي والتلقائي لرفع ملف التجسس هذا من جذوره أثناء التنصيب التلقائي للويندوز والبرامج.

```
reg delete HKLM\SOFTWARE\Microsoft\Internet Explorer\Extensions\{c95fe080-8f5d-11d2-a20b-00aa003c157a} /f
```

وإلى التفاصيل ولكنني سوف لن أعيد هنا شرح الأوامر التي قمت بشرحها في المثال السابق، بل سأقوم بشرح ما هو جديد منها. ما هو جديد هنا من أوامر هو أمر الحذف أوالمسح .deleteأما المسار في الريجستري فيجب أن يختلف عن المثال السابق، ولكن يجب أن نتوقف ونتأمل قليلاً ثم نسأل أنفسنا أين هو المفتاح الذي يحتوي على ملف التجسس Alexa Spyware، أقول هو المفتاح المعقد { .c95fe080-8f5d-11d2-a20b-00aa003c157aنحن تكلمنا قبل عدة حصص عن CLSID وقلنا كيف هو مقعد ويحوي على معلومات أكثر من برنامج

يدخل في صنعه. وملف التجسس هذا هو أحد الأمثلة المعقدة على CLSID.وبما أن ملف التجسس هذا يكون المستفيد منه عدة أماكن لا يعلمها إلى الله سبحانه وتعالى وحتى لا ينكشف الأمر بسهولة تمت برمجته لكي يكون بهئية CLSID إضافي إلى متصفح إكسبلور. فلو كان إسم ملف التجسس في الريجستري تحت عنوان Alexa Spyware لكان صيداً سهلاً للتعرف عليه ولكانت مفاتيحه في الريجستري كثيرة يسهل التعرف عليها وتشخيصها أما إذا كان بهئية مفتاح واحد وعلى شكل CLSID كما هو الواقع فيصبح من الصعب تشخيصه والتعرف عليه، ومع كل هذا التعقيد تم التعرف عليه وتشخيصه ورفعه من جذوره.

3-تصدير مفتاح معين. بالإمكان كتابة أمر تنفيذي يقوم تلقائياً بتصدير مفتاح معين من الريجستري قبل إجراء أي تغيير على المفتاح نفسه بأمر تنفيذي آخر. دعنا نصدر مفتاح الريجستري الذي يحوي على القيمة الإفتراضية التي يحجزها الويندوز لسلة المهملات أثناء التنصيب. القيمة الإفتراضية التي يحجزها الويندوز لسلة المهملات تـأخذ من حجم القرص الصلب وتكون قيمتها دائماً 10% من حجم القرص الصلب. تعتبر هذه النسبة في نظر الكثيرين غير واقعية، ولكن لماذا؟ فنسبة 10% من قرص صلب حجمه 120 جيجابايت تعادل 12 جيجابايت، ونسبة 10% من قرص صلب حجمه 80 جيجابايت تعادل 8 جيجابايت، ، ونسبة 10% من قرص صلب حجمه 40 جيجابايت تعادل 4 جيجابايت، وهكذا. ولكن هل تحتاج سلة المهملات لمثل هذه الحجوم، الجواب كلا ثم كلا. ممكن أن تحتاج سلة المهملات لحجم 1 جيجابايت على الأكثر، لأن وجود ملفات بحجم 1 جيجابايت أو من هذا القبيل في سلة المهملات يؤثر سلباً على أداء الويندوز وبخاصة الإقلاع ويؤثر كذلك على برامج تسريع القرص الصلب، لذا يجب التخلص منها بأقرب وقت ممكن. وكلما كبر حجم القرص الصلب كلما إزداد الأمر سوءاً وذلك لزيادة سعة سلة المهملات أي زيادة الحجم الضائع من القرص الصلب من دون أي ضرورة تذكر.
الأمر التنفيذي الذي يقوم بتصدير مفتاح الريجستري الخاص بالقيمة الإفتراضية لسلة المهملات هو:

```
reg export
HKLM\SOFTWARE\Microsoft\Windows\CurrentVers
ion\Exp lorer\BitBucket\Percent
```

%systemdrive%\RecyBinSZ.reg

التفاصيل: الأمر الجديد هنا هو تصدير .export المفتاح هو Percent، ملف التصدير RecyBinSZ.reg يكون ملف ريجستري عادي يقوم مستخدم الحاسوب بإختيار إسمه بشرط أن ينتهي الإسم بالإمتداد reg، أما %systemdrive% فتشير إلى القرص الصلب الذي يحوي على الويندوز والذي سيتم خزن ملف التصدير فيه بغض النظر عن حرفه سواء أكان C أو D أو E. أي بمعنى آخر أن %systemdrive%\RecyBinSZ.reg يمثل المسار الذي سيتم خزن ملف التصدير فيه على القرص الصلب الحاوي على الويندوز. ومما تجدر الإشارة إليه أنه من الممكن تغير حجم سلة المهملات يدوياً، وكالتالي: إضغط بيمين الماوس على سلة المهملات ثم قم بإختيار آخر إحتمال Properties من القائمة التي ستظهرز تابع تسلسل الصورة التالية.

هذه هي أهم الأوامر التي تستخدم في التنصيب التلقائي. هناك أوامر أخرى ولكنها لا تستخدم في التنصيب التلقائي مثل طباعة مفتاح معين من الريجستري، ومقارنة المفاتيح وغيرها لكني سأكتفي بهذا الحد خشية تعقيد الموضوع أكثر.

تحسين أداء NTFS

يتأثر نظام NTFS لتهيئة القرص الصلب بعدة عوامل، منها حجم العنقود Cluster size، مستوى تبعثر الملفات Fragmentation level، وكذلك البرامج المنصبة ولا سيما برامج الوقاية من الفايروسات والأمان. بالإضافة إلى ذلك، فإن الخدمات التي يقدمها NTFS مثل كبس المجلدات Folder

compression و الفهرسة Indexing هي الأخرى لها دور بارز في التأثير على أداء NTFS. وفي هذه الحصة سنقوم إن شاء الله تعالى بإستعراض الخطوط العريضة التي من شأنها تحسين أداء NTFS.

حجم العنقود Cluster size

لتحديد الحجم الإفتراضي لعنقود NTFS، فمن المفترض وقبل أن يقدم كل منا على عمل تهيئة للقرص الصلب بنظام NTFS أن يتأمل ويفكر قليلاً ويخمن بعقله حجم كل ملف من الملفات التي سيقوم بخزنها في ذلك القرص الصلب بعد تهيئته. فمن الأسئلة التي قد تخطر على البال في تلك اللحظة:

· هل أن جميع الملفات لها نفس الحجم تقريباً

· هل سيحافظ كل ملف على حجمه الأصلي أم سيكبر ويكبر مع مرور الزمن

· هل أن حجم أي ملف من الملفات المراد خزنها أو تنصيبها أصغر من الحجم الإفتراضي للعنقود

عند تهيئة القرص الصلب بنظام NTFS وبالطريقة العادية أي من دون أي يتدخل المرء بتحديد حجم العنقود بل النظام هو الذي سيحدد حجم العنقود تلقائياً، فإن الحجم الإفتراضي للعنقود سيكون 4 كيلوبايت. فإذا كنت تعتقد بأن جميع ملفاتك ستكون أصغر من الحجم الإفتراضي للعنقود أي أصغر من 4 كيلوبايت كما أنها لا تكبر في الحجم مع مرور الزمن، فالحجم الإفتراضي للعنقود (4 كيلوبايت) يكون مناسباً تماماً. ولكن ما الذي سيحدث لو بقى الحجم الإفتراضي العنقود على 4 كيلوبايت وكانت الملفات المراد تنصيبها على القرص الصلب أكبر من 4 كيلوبايت بالإضافة إلى أن حجمها سيزداد مع مرور الحجم؟ أقول كل ملف حجمه أكبر من الحجم الإفتراضي للعنقود سينتشر فوق عدد أكبر من العناقيد وبذلك يزداد تبعثر الملفات أي يحدث ال Fragmentation كما وستزداد الخسارة في الحيز الضائع من القرص الصلب وهذه حقائق لا مفر منهما أبداً. وكمثال على هذه الحالة هي حالة تهيئة القرص الصلب المعد لتنصيب الويندوز والبرامج. فلا ملفات الويندوز ثابتة الحجم ولا ملفات البرامج ثابتة الحجم. لذا يفضل بل يوصى به مراعاة الحجم

الإفتراضي للعنقود عند عمل تهيئة للقرص الصلب بنظام NTFS. الإحترافيون يفضلون أن يكون الحجم الإفتراضي للعنقود إما 16 كيلوبايت أو 32 كيلوبايت إعتماداً على نوع الإستخدام وحجم البرامج المنصبة. فالكمبيوتر الذي يحوي على برامج مثل Photoshop و AutoCAD وغيرها من برامج الرسم والتصميم الثقيلة يفضل جعل الحجم الإفتراضي للعنقود 32 كيلوبايت. فلربما لاحظ اللذين يمتلكون مثل هذه البرامج بعض البطئ عند تشغيلها، وأحد أسبابها هي صغر الحجم الإفتراضي للعنقود بالمقارنة مع حجم ملفات هذه البرامج مما يجعل ملفاتهما مبعثرة هنا وهناك والتي تستلزم التسريع بين الحين والآخر لغرض تحسين سرعة إقلاعها. ملاحظة: عند القيام بزيادة الحجم الإفتراضي للعنقود من 4 كيلوبايت إلى 16 أو 32 كيلوبايت، فإن خدمة الكبس أو الضغط التي يوفرها NTFS مجاناً تعتبر لاغية، أي ان خدمة الكبس تكون موجودة فقط عندما يكون الحجم الإفتراضي للعنقود 4 كيلويابت.

مشكلة NTFS والأسماء القصيرة للملفات 8.3 Short File Names في كل مرة تقوم بإنشاء أو نسخ ملف طويل الإسم (أي عدد أحرف إسمه أكثر من 8 ما عدا الإمتداد)، فإن NTFS وبنفس الوقت يقوم بإنشاء إسماً قصيراً مرادفاً عدد أحرف أسمه 8 مع الإحتفاظ بنفس الإمتداد الأصلي (قاعدة أسماء نظام DOS) لذلك الملف. تخيل المسألة عندما يكون هناك عشرات الآلاف من الملفات بإسماء طويلة وكم سيستغرق NTFS لإنشاء أسماء قصيرة مرادفة لها. والأمر الأدهى والأمر من كل هذا، أن NTFS يستخدم الأحرف الستة الأولى من الإسم الطويل كأساس لعمل الإسم القصير، فإذا كان لديك أكثر من ملف بإسماء طويلة تكون بدايتها نفس الأحرف الستة، فهنا تحدث المشكلة ويحصل التعارض الذي قد يؤدي إلى التعليق في كثير من الأحيان. ولتلافي هذه المشكلة وكذلك لزيادة أداء NTFS لابد من تعطيل هذه الخدمة الغير الضرورية تماماً، أي خدمة إنشاء إسماء قصيرة مرادفة للأسماء الطويلة. هناك مفتاح في الريجستري يقوم بتعطيل هذه الخدمة ومساره وقيمته هما

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlS

et\Contro l\FileSystem
"NtfsDisable8dot3NameCreation "=dword:00000001

وللأسف أن القيمة الإفتراضية للمفتاح
NtfsDisable8dot3NameCreation والتي يقوم الويندوز بتعينها أثناء التنصيب هي 0 وتعني إنشاء أسماء قصيرة مرادفة للأسماء الطويلة، أما القيمة 1 (كما تراه أعلاه) يعني تعطيل هذه الخدمة. يجمع الكثيرون على أهمية تعطيل هذه الخدمة من أجل زيادة أداء NTFS ولكن القرار الأول والأخير يعود لمستخدم الحاسوب.

مشكلة NTFS وتركيبة المجلدات والملفات

من أحد الأسباب التي دعت إلى ظهور نظام NTFS هو مقدرته على التعامل مع الأحجام الكبيرة للأقراص الصلبة الحاوية على عدد كبير جداً من المجلدات والملفات. ولكن مع كثرة تشعب المجلدات (الواحد بداخل الآخر) وغزارة الملفات فيها، ستخفض تلك التركيبة من أداء NTFS ويجعل فتح المجلدات بطيئاً. وللحصول على أداء متميز ل NTFS، يفضل عند إنشاء أو صنع مجلد مراعاة القواعد التالية:

· تجنب وضع عدد كبير من الملفات في مجلد واحد قد الإمكان وبخاصة إذا كانت هذه الملفات تستخدم بشكل دائم وسريع من قبل البرامج. بل يفضل هنا تقسيم الملفات على عدد من المجلدات، فبدلاً من أن يتركز الضغط على مجلد واحد، سينقسم الضغط اعلى عدد المجلدات (أي يخف الضغط) وبالتالي يتحسن الأداء ومن المفترض أن تفتح المجلدات بشكل أسرع.

· إذ كان من غير الممكن تقسيم الملفات على عدد من المجلدات، إذاً يجب في هذه

الحالة تعطيل خدمة إنشاء أسماء قصيرة مرادفة للأسماء الطويلة التي تكلمنا عنها أعلاه، للحفاظ على ماء الوجه على أقل تقدير. وكقاعدة ثابتة، الزمن الذي تستغرقه برامج تصليح وتسريع القرص الصلب يكون عادة أطول عند فحص مجلد بملفات كثيرة بالمقارنة مع الزمن المستغرق لفحص مجلد بعدد أقل من الملفات.

لكل مجلد أوملف متواجد على ظهر قرص صلب مهيء بنظام NTFS له راية فيها معلومات مدونة عن تأريخ وزمن آخر إستخدام أو دخول لذلك الملف أوالمجلد. تخزن معلومات كل راية في الذاكرة في بادئ الأمر ثم تخزن في النهاية في القرص الصلب في مقطع MFT الذي تكلمنا عنه أعلاه في الحصة السابقة. والمعلومات المدونة في كل راية تحدث تلقائياً كل ساعة بشرط أن يكون الجهاز شغالاً بشكل مستمر ولمدة أكثر من ساعة. وطالما أن معلومات الراية تحدث تلقائياً بكل ساعة من العمل المتواصل، إذاً ليس من الضرورة أن تكون تلك المعلومات المدونة آخر أو أدق المعلومات التي تعكس حال إستخدام المجلد أو الملف. والذي يهمنا من هذه المسألة برمتها، هو هل أن عمل هذه الراية وتدوين المعلومات فيها ضروري، وهل لها تأثير على أداء النظام؟ أقول ضررها أكبر بكثير من نفعها عندما يحتوي القرص الصلب على عدد كبير من الملفات. فتخيل قرص صلب وفيه آلاف الملفات ومئات المجلات وتحتاج كلها إلى تدوين معلومات، وإستخدام ذاكرة وقرص صلب لخزنها وتأثيرها النهائي على زيادة بعثرة الملفات على القرص الصلب. ولحسن الحظ، هنلك مفتاح في الريجستري يمكن تغيير قيمته الإفتراضية من 0 إلى 1 لكي يقوم بتعطيل هذه الخدمة الغير الضرورية (بنظر وإجماع الكثيرين). مسار هذا المفتاح وقيمته هي نفس مسار وقيمة المفتاح الذي سبقه والإختلاف في عنوان المفتاحين تكون كالأتي:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Contro l\FileSystem "NtfsDisableLastAccessUpdate "=dword:00000001

لقد وجد أن تعطيل هذه الخدمة يساهم وبشكل واضح في تحسين الأداء بغض النظر عن مواصفات الجهاز.

NTFS وخدمة كبس المجلدات Folder Compression

ويندوز إكسبي جاء ومعه مدمجاً خدمة مجانية متواضعة لكبس وفك الكبس عن الملفات والمجلدات. هذه الخدمة البسيطة تكون شغالة على الدوام طالما أن الويندوز شغال مستهلكة بعض من مصادر النظام System resources في الوقت الذي يعتبر من النادر إستخدام هذ الخدمة لا سيما وأن معظمنا إن لم نقل جميعنا يستخدم برامج كبس أخرى إحترافية مثل Winzip أو Winrar وكثير غيرها. ومما تجدر الإشارة إليه أن كلما زادت نسبة تحرير المصارد للنظام كلما إزداد أداء وإستقرار النظام، بمعنى آخر أنه كلما قلت المصادر المتوفرة للنظام كلما ضعف الأداء وأصبح أقل إستقرارية. والله وحده يعلم، بأن تحرير مصدر واحد للنظام قد يكون سبباً كافياً في إستقراريته ومن ثم تحسن أداءه. وكم مرة كان ملف واحد أو برنامج واحد صغير سبباً في عدم إستقراية النظام. بالإضافة إلى ذلك فلقد وجد أن بقاء هذه الخدمة شغالة يبطئ من عمل برامج الكبس الإحترافية. كما وجد أيضاً أن هذه الخدمة يضعف أداءها بشكل رهيب عند الكبس وفتح الكبس عن المجلدات الكبيرة الحجم. لذا أجمع الكثيرين على تعطيلها ولكن يبقى القرار النهائي لك عزيزي القارئ. وأسهل طريقة لتعطيل خدمة كبس المجلدات في الريجستري هي بالذهاب إلى زر إبدأ Start، ثم إختيار تشغيل Run ثم كتابة الأمر التالي

regsvr32 /u zipfldr.dll

نحن قلنا في حصة سابقة عندما تكلمنا عن دور الريجستري في التنصيب التلقائي للويندوز والبرامج من أن البرنامج المساعد regsvr32 يستخدم لتسجيل الملفات التي لها إمتداد dll في الريجستري، والآن قد عاد إليكم هذا البرنامج المساعد من جديد ولكن بعملية معاكسة وهي القيام برفع التسجيل عن الملف zipfldr.dll من خلال إضافة الأمر /u. فرفع التسجيل عن الملف zipfldr.dll يعني تعطيله وبذلك يصبح مهملاً في مجلد system32.

ولمن أراد إعادة هذه الخدمة إلى حالتها الإفتراضية أي إلى وضع التمكين فسيكون الأمر

regsvr32 zipfldr.dll

قد يتساءل البعض هنا هل سيأثر رفع خدمة كبس/فك كبس المجلدات على تنصيب البرامج المندمجة في سيدي الويندوز أو تحديثات الويندوز لأن كل هذه البرامج والتحديثات تكون مضغوطة؟ أقول كلا لأن كل هذه البرامج والتحديثات تستخدم برنامج مساعد آخر مندمج في الويندوز يدعى MakeCAB للكبس اوبرنامج مساعد آخر يدعى Expand لفك الكبس عنها.

خدمة الفهرسة Indexing service

إحدى أهم الأسباب التي تجعل ويندوز إكسبي يحتفظ بمعلومات مدونة عن الملفات هو لغرض تسهيل وتسريع عملية البحث في القرص الصلب. ويمكن تشبيه خدمة البحث بوجود خدمة الفهرسة كخدمة البحث السريعة Find Fast في MS OFFICE. ولكن عملية الفهرسة هذه لها سلبياتها أيضاً، فهي تستهلك جزء لا يستهان به من مصادر النظام، كما أنها تستهلك جزءاً من وقت المعالج والقرص الصلب والذاكرة لانها خدمة تعمل طوال الوقت طالما أن الجهاز شغال، وبالتالي

تكون سلبياتها أكثر بكثير من إيجابياتها. قد يتساءل البعض، لو قمنا بتعطيل خدمة الفهرسة فهل سنفقد خدمة البحث؟ أقول كلا، فخدمة البحث لا يعتمد عملها على عمل خدمة الفهرسة وإنما عند تشغيل خدمة البحث وكانت خدمة الفهرسة موجودة فسيكون البحث أكثر سرعة ولربما أكثر دقة! ولكن كم منا يحتاج لخدمة البحثفي الساعة أو في اليوم من عمله على الجهاز؟ فلو أجرينا مقارنة سريعة بين حاجتنا لخدمة البحث من جهة وبين حاجتنا لتحرير مصادرللنظام، وتخليص المعالج من عمل تدوين المعلومات المستمر بالإضافة إلى إخلاء الجزء المشغول من القرص الصلب والذاكرة الذي تحتله معلومات التدوين من جهة أخرى لوجدنا أن إلغاء خدمة الفهرسة أفضل بكثيرمن بقاءها. وأنا شخصياً أفضل إلغاء خدمة الفهرسة لأنها ومن دون أي شك ستساهم في زيادة أداء النظام بشكل عام ولكن القرار النهائي يعود إليك عزيزي القارئ.

هناك طريقتان لتعطيل خدمة الفهرسة في ويندوز إكسبي. الطريقة التلقليدية، إنقر مرتين على أيقونة My Computer، ثم إنقر بيمين الماوس على أيقونة القرص الصلب وقم بإختيار خصائص Properties في أسفل القائمة، ستحصل على النافذة التالية

لكل قرص صلب مهيئ بنظام NTFS خدمة فهرسة لذا يتوجب إلغاء هذه الخدمة من كل الأقراص لمن أراد. أما الطريقة الثانية لتعطيل خدمة الفهرسة فهي عن طريق الريجستري، وذلك من خلال مسار وقيمة المفتاح التالي

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Servic es\cisvc
"Start "=dword:00000004

علاقة أداء النظام بالذاكرة الظاهرية Paging File

عندما يكون لديك جهاز تكون فيه ذاكرة النظام أقل من 256 ميجابايت ولم يتم تخفيف عدد الخدمات والبرامج التلقائية التشغيل الغير الضرورية ثم تقوم في نفس الوقت بعدة مهام فمن الطبيعي أن يزداد إستهلاك الويندوز للذاكرة وفي كثير من الأحيان تصل حاجة الويندوز إلى الذروة بحيث أنه لم يتبقى أي شيء من ذاكرة النظام إلا وتم إستهلاكها أي تستنفذ ذاكرة النظام بأكملها؟ ومما يزيد من إستهلاك ذاكرة النظام الحقيقة أكثرهي أن لبعض لوحات الأم كرافيكس مدمج يستلهم ذاكرته من ذاكرة النظام، فهذا ومن دون أي شك يزيد المسألة أكثر سوءاً وتعقيداً. إذاً ماذا سيحدث، هل سيحدث التعليق، وهل سيعيد الويندوز تشغيل نفسه تلقائياً، وهل ستخسر كل عملك ونتاجك إلى تلك اللحظة إن لم تكن قد قمت بخزنه وغيرها من الأسئلة الكثيرة التي لربما تخطر على البال؟ الجواب على هذه الأسئلة سيكون بكلا ولكن بشرط أن يكون الجهاز سليم ومعافى من مشاكل أخرى غير الذاكرة. فأثناء تنصيب الويندوز، يقوم محرك التنصيب بصنع ذاكرة ظاهرية أو إضافية سميها ما شئت على القرص الصلب، ومن الآن فصاعداً سنسميها بالذاكرة الظاهرية لأن ظاهرها ذاكرة ولكن حقيقتها هي حيز مقطوع أو مأخوذ من القرص الصلب. وفي مصطلحات الكمبيوتر يطلق على هذه الذاكرة الظاهرية بإما Virtual Memory أو Paging file. مقدار أو حجم هذه الذاكرة الظاهرية الذي سيتم حجزه من القرص الصلب أثناء تنصيب الويندوز يكون دائماً 1.5 مضروباً بحجم ذاكرة النظام الحقيقة. فمثلاً لو كان حجم ذاكرة النظام الحقيقة 256 ميجابايت، لأصبح حجم الذاكرة الظاهرية 384 ميجابايت وهكذا. وعندما تقوم بعدة مهام ويشعر الويندوز بأن ذاكرة النظام الحقيقة شارفت على الإستنزاف ويحتاج إلى ذاكرة إضافية لتسيير المهمة الجارية تنفيذها ولتفادي حدوث توقف أو تعليق للنظام يقوم الويندوز بإستخدام الذاكرة الظاهرية. سرعة الذاكرة الظاهرية هي نفس سرعة القرص الصلب أي بضعة أجزاء من الألف من الثانية على أحسن تقدير، بينما سرعة الذاكرة الحقيقة للنظام هي بضغة أجزاء من البليون من الثانية. بمعنى آخر أن الذاكرة الحقيقة أسرع من الذاكرة الظاهرية بمليون مرة تقريباً!. لاحظ الفرق في سرعة الذاكرتين. ولهذا السبب تعاني الأجهزة القليلة الذاكرة الحقيقة للنظام من البطئ مهما بغلت سرعة المعالج. ولنفس السبب ينصح دائماً وأبداً بأن تكون ذاكرة النظام وبالأخص لويندوز إكسبي بما لا يقل عن 512 ميجابايت للحصول على أفضل أداء للنظام لأن في هذه الحالة سيقل إعتماد الويندوز على الذاكرة الظاهرية بشكل

كبير ويكاد يكون معدوماً عندما تكون الذاكرة الحقيقة للنظام أكبر من 512 ميجابايت. وعندما تكون لديك ذاكرة نظام حقيقة مقدارها 512 ميجابايت أو أكثر فبإمكانك تعطيل الذاكرة الظاهرية. هناك طريقتان لتعطيل الذاكرة الظاهرية، ما يخصنا هنا تعطيل الذاكرة الظاهرية عن طريق إعدادات الريجستري.وعليه سيكونمسار المفتاح وقيمته في الريجستري لتعطيل الذاكرة الظاهرية:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
Session Manager\Memory Management]
“DisablePagingExecutive”=dword:00000001
```

القيمة الإفتراضية لهذا المفتاح أثناء تنصيب الويندوز هي 0 وتعني وجود الذاكرة الظاهرية أما عند تغير القيمة إلى 1 (كما هو الحال أعلاه) فسيتم يتعطيل الذاكرة الظاهرية لمن كانت له ذاكرة حقيقة للنظام مقدارها 512 ميجابايت أو أكثر. والفائدة الرئيسية من تعطيل الذاكرة الظاهرية هو لزيادة أداء وإستقرارية النظام وبشكل واضح ولربما ملفت للنظر.
ومما تجدر الإشارة إليه إلى أنه هناك طرق عديدة لتحسين أداء النظام بوجود الذاكرة الظاهرية أي عندما تكون ذاكرة النظام الحقيقة قليلة، ولكني لا أريد التطرق إليها لأنها ستخرجنا كثيراً عن صلب الموضوع الذي نتكلم عنه. ومهما تكلمنا عن أفضل هذه الطرق لتحسين أداء النظام وذلك بتحسين عمل الذاكرة الظاهرية يبقى خيار ترقية الذاكرة الحقيقة للنظام هو السبيل الأمثل والأفضل لتحسين أداء وإستقرارية النظام ولا سيما في وقتنا الحاضر حيث تتوفر الذاكرة بأنواع وسرع مختلفة وبأسعار تعتبر منخفضة إن لم نقل بأسعار زهيدة.

تحسين الأداء من خلال نقل ملف النظام إلى الذاكرة
هناك إعداد بسيط في الريجستري يمكن عمله لتحسين أداء النظام بشكل كبير جداً. وبموجب هذا الإعداد يتم حجز مقعد أولي من ذاكرة النظام الحقيقة مقداره 4 ميجابايت قابلاً للزيادة إلى حد 8 ميجابايت وذلك لتحميل وتسريع ملف النظام أو ما يسمى XP Kernel أو NT Kernel. في الحالات الإعتيادية أي عندما يكون إعداد الريجستري الذي نتكلم عنه معطلاً، فإن ملف النظام يبقى في القرص الصلب. أما عندما نقوم بتفعيل هذا الإعداد في الريجستري، فعند كل إقلاع للويندوز وقبل

ظهور شاشة سطح المكتب سيتم نقل ملف النظام من القرص الصلب إلى مقعد الذاكرة الذي تم حجزه. ولكن ما الفائدة من عمل ذلك وخسارة على الأقل 4 ميجابايت من الذاكرة الحقيقة للنظام، أقول تفعيل هذا الإعداد سيجعل سرعة إستجابة ملف النظام لعمليات تبادل المعلومات (قراءة/كتابة) أسرع بما لا يقل عن أربعين مرة بالمقارنة مع بقاء ملف النظام في القرص الصلب. بمعنى آخر، عندما يبقى ملف النظام في القرص الصلب، وفي أحسن الأحوال وذلك عندما يكون هناك أسرع قرص صلب وأسرع معالج فلا يتجاوز معدل سرعة تبادل المعلومات عن 40 ميجابايت لكل ثانية. ولكن عند رفع ملف النظام إلى الذاكرة، فسيرتفع معدل تبادل الملعومات إلى أكثر من جيجابايت لكل ثانية وهذه حقيقة. لتفعيل هذا الإعداد في الريجستري يتطلب أن تكون ذاكرة النظام الحقيقة على ألاقل 256 ميجابايت ويفضل بل يؤكد أن تكون أكبر من ذلك.
مسار المفتاح وقيمته في الريجستري والذي يقوم بحجز مقعد ما لا يقل عن أربعة ميجابايت من الذاكرة الحقيقة لملف النظام ومن ثم تولي نقله أثناء الإقلاع إلى هذا المقعد من الذاكرة الحقيقة هو نفس مسار المفتاح السابق

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
Session Manager\Memory Management]
“LargeSystemCache”=dword:00000001
```

إحدى النقاط السلبية التي تم كشفها لحد الآن نتيجة تفعيل هذا الإعداد، هو أن معظم بطاقات الكرافيكس التي تقوم شركة ATI وبالأخص محركات Radeon فقط لا تتوافق مع تفعيل هذا الإعداد لذا ينصح بعدم تفعيل هذا الإعداد لمن يملك مثل هذه الأنواع من بطاقات الشاشة. بمعنى آخر ليس هناك أي ضرر من تفعيل هذا الإعداد في حالة وجود أنواع أخرى من بطاقات الشاشة أي غير .ATI Radeon

تحسين الأداء من خلال زيادة كفاءة معدل القراءة والكتابة من وإلى القرص الصلب وملف النظام
الحل في الريجستري وعلى المسار التالي

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Contro l\Session Manager\Memory Management

وكان هدفي في الريجستري هو المفتاح .IoPageLockLimitوظيفة هذا المفتاح هو تحديد حيز من الذاكرة لكي يستخدمها ملف النظام Kernel لتسريع عمليات نقل المعلومات من وإلى القرص الصلب. الحالة الإفتراضية إما أن يكون هذا المفتاح غير موجود أصلاً (ولا أعرف السبب) أو أن يكون هذا المفتاح موجوداً ولكن بدون قيمة محددة أي بمعنى تترك قيمته للويندوز لكي يحددها متى يشاء ومتى تظهر الحاجة إليها. فمن المؤكد حاجة ملف النظام لهذا المفتاح وبشدة أثناء العمليات التي تتطلب قراءة وكتابة شديدة مثل النسخ لسيدي والديفيدي والمسح الضوئي Scanning وتشغيل الألعاب وغيرها. فعدم وجود هذا المفتاح مشكلة كبيرة ووجوده بقيمة يحددها الويندوز متى يشاء مشكلة أخرى لا تقل سوءاً عن سابقتها. الجدول التالي يظهر القيم المثالية التي يمكن تثبيت قيمة هذا المفتاح عليها في الريجستري والتي تعتمد على ذاكرة النظام الحقيقة:

ذاكرة النظام الحقيقة	IoPageLockLimit	
MB	Decimal	Hex
128	16384	4000
192	36864	9000
256	65536	10000
384	98304	18000
512	131072	20000
768	196608	30000
1024	262144	40000
1536	393216	60000
2048	524288	80000

فمثلاً: عندما تكون ذاكرة النظام 256 يأخذ المفتاح القيمة التالية

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
Session Manager\Memory Management]
"IoPageLockLimit"=dword:00010000
```

وعندما تكون ذاكرة النظام 512 يأخذ المفتاح القيمة التالية

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
Session Manager\Memory Management]
"IoPageLockLimit"=dword:00020000
```

المشكلة تتعلق بالعواقب التي قد تترتب على بقاء عمليات القراءة والكتابة من دون تحسين أداء. فتخيل كم هو تأثير دقيقة زائدة من العمل المتواصل الشديد على درجة حرارة المعالج وبخاصة إذا كان التبريد ليس بالمستوى المطلوب، فمثلاً تخيل إقلاع الويندوز الذي يعتبر من العمليات المعقدة والشديدة، فخلال إقلاع الويندوز ترتفع درجة حرارة المعالج بأكثر من 20 درجة مئوية لمعالجات إنتل وأكثر من 25 درجة مئوية لمعالجات AMD، علماً بأن مدة الإقلاع الجيدة قد لا تستغرق أكثر من 30 ثانية فلاحظ عزيزي القارئ جذور المشكلة. وكمثال آخر تخيل عندك معلومات كبيرة تقدر ب 40 جيجابايت في قرص صلب وتريد نقلها إلى قرص آخر فكم ستستغرق عملية النقل من دون تحسين وبوجود التحسين على القراءة والكتابة؟

فلو تأملنا قرصين سريعين معدل نقل المعلومات من وإلى الآخر يكون بحدود 40 ميجابايت لكل ثانية من دون وجود تحسين على القراءة والكتابة، ولو إفترضنا أن معدل النقل سيتحسن ويصل إلى 50 ميجابايت لكل ثانية (وهذا أقل شيء متوقع) بعد تعديل قيمة المفتاح أعلاه بحسب الجدول. إذن سيكون الزمن المستغرق لنقل المعلومات في الحالة الأولى (من دون تحسين) بحدود 17 دقيقة، بينما سيكون

الزمن المستغرق لنقل المعلومات في الحالة الثانية (بوجود التحسين) بحدود 13 دقيقة أو أقل. فأربعة دقائق من العمل الشاق والمتواصل لنقل المعلومات تعتبر طويلة. والأمر سيزداد سوءاً أكثر إذا قمت بأكثر من عمل معقد وشديد في آن واحد من دون عمل التحسينات التي ذكرناها. فمثلاً عندما تقوم بعمل scanning وتعمل أيضاً نقل للمعلومات من قرص لآخر، فهاتين العمليتين تعتبران من العمليات المعقدة الشديدة، ومن دون عمل التحسينات على القراءة والكتابة ومهما كان عندك من ذاكرة نظام فسيعاني النظام من البطئ الشديد وقد يصل به المطاف إلى حد التعليق.

تعطيل خدمة التشغيل التلقائي عند حدوث الخلل الجذري وظهور الشاشة الزرقاء

Disable Automatic Restart in the event of a BSOD

إفتراضياً، يقوم ويندوز بإعادة تشغيل نفسه تلقائياً عند حدوث الخلل الجذري Fatal error وظهور الشاشة الزرقاء. ونحن هنا لسنا بصدد الحديث عن أسباب ظهور الشاشة الزرقاء عند بدأ التشغيل ولا مناقشة حلولها فهذا موضوع آخر طويل ومعقد. في الحقيقة هناك إختلاف كبير بين حدوث الخطأ الجذري أثناء الإغلاق وبين حدوث الخطأ الجذري في حالة تشغيل برنامج معين أو تعارض بين البرامج ومشغلات الأجهزة. فمن الطبيعي عند حدوث مشكلة ما عند الإقفال إما أن يعيد الويندوز تشغيل نفسه بدلاً من الإقفال أو أن يطول أمد الأقفال وفي بعض الأحيان يتوجب عمل الإقفال إجبارياً. أما إذا كنت في منتصف عمل ما، وحدث خطأ جذري ما بحيث يؤدي إلى حدوث إعادة التشغيل التلقائي للويندوزمن دون سابق إنذار، فهو يمثل خسارة وبخاصة إذا إستمر الحال على هذا المنوال لعدة مرات. وللتخلص من خدمة التشغيل التلقائي للويندوزعند حدوث الخطأ الجذري فهذا مسار المفتاح في الريجستري وقيمته التي تعطل هذه الخدمة التي يجمع الكثيرين على وجوب تعطيليها

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\CrashControl]
“AutoReboot” =dword:00000000
```

****

****

****

تعطيل خدمة التحكم بالريجستري عن بعد Disable Remote Registry Service

إفتراضياً تكون هذه الخدمة فعالة أو شغالة، وبموجبها يكون بإمكان مستخدم الكمبيوتر من عمل تعديل على ريجستري جهازه المنزلي في الوقت الذي يكون هو في العمل، وهذا هو عنوان هذه الخدمة (التحكم بالريجستري عن بعد). فإذا كنت فعلاً تريد عمل تعديلات على ريجستري جهازك المنزلي وأنت بعيد عنه وفي نفس الوقت تكون متأكداً من أن جهازك لن يخترق ولن يعبث أحد بريجستري جهازك المنزلي وأنت على إتصال بالإنترنت، فأترك هذه الخدمة شغالة. أما في كل الحالات الأخرى فيجب تعطيل هذه الخدمة لخطورة بقاءها شغالة وإلا أي إختراق وعبث في الريجستري لربما يسبب خراب كامل للنظام وفي حينها لا ينفع معه إلا تنصيب جديد للويندوز (اللهم إلا إذا كان عندك نسخة إحتياطية جيدة من الريجستري). مسار المفتاح في الريجستري وقيمته التي تعطل خدمة التحكم بالريجستري عن بعد

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RemoteRegistry]
“Start”=dword:00000004
```

****

****

****

تعطيل خدمة التحكم والمساعدة بالجهاز المنزلي عن بعد Disable Remote Assistance and Remote Desktop

وهذه الخدمة تشبه تماماً خدمة التحكم بالريجستري عن بعد. إفتراضياً تكون هذه الخدمة فعالة أو شغالة، وبموجبها يكون بإمكان مستخدم الكمبيوتر من تشغيل البرامج والطباعة وغيرها على جهازه البيتي في الوقت الذي يكون هو في العمل. المشكلة الكبيرة في هذه الخدمة كسابقتها هي مشكلة توفير الأمان اللازم للجهاز البيتي من الإختراق أثناء التحكم به عن بعد. فإذا كنت لا تحتاج إلى الدخول إلى

جهازك البيتي وأنت في عملك أو أي مكان آخر فالمفروض أن تقوم بتعطيل هذه الخدمة التي تعتبر في هذه الحالة غير ضرورية بل ضارة إن بقت فعالة. لتعطيل هذه الخدمة بالكامل نحتاج إلى ثلاثة مفاتيح ولكن بمسار واحد في الريجستري

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server]
“AllowTSConnections”=dword:00000000
“fAllowToGetHel”=dword:00000000
“fDenyTSConnections”=dword:00000001
```