

# Začněte skutečně ovládat PC. Používejte příkazovou řádku



Jakub Čížek  
10. dubna 2014

Zkušený uživatel počítače by měl zvládnout nejen základy programování, ale stejně tak by se měl vyznat v příkazové řádce – textové konzoli. I ta na Windows umí skvělé kousky.

```
create [/S system [/U username [/P [password]]]
        [/M [password]] /SC schedule [/MO modifier] [/D day]
        [/Hs] [/I idletime] /TN taskname /TR taskrun [/ST starttime]
        /Serval] [ /RT endtime [ /DU duration] [/K] [/XML xmlfile] [/VI]]
        /rtdate] [/ED enddate] [/IT] [/NP] [/Z] [/F] [/HRESULT] [/P]]

an administrator to create scheduled tasks on a local or
system.

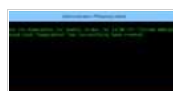
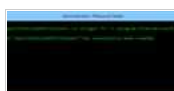
ut:
tem Specifies the remote system to connect to. If omitted
the system parameter defaults to the local system.

ername Specifies the user context under which SchTasks.exe
should execute.

ssword] Specifies the password for the given user context.
Prompts for input if omitted.

ername Specifies the "run as" user account (user context)
under which the task runs. For the system account,
valid values are "", "NT AUTHORITY\SYSTEM"
or "SYSTEM".
For v2 tasks, "NT AUTHORITY\LOCALSERVICE" and
"NT AUTHORITY\NETWORKSERVICE" are also available as well
as the well known SIDs for all three.

ssword] Specifies the password for the "run as" user.
To prompt for the password, the value must be either
```



**30 FOTOGRAFIÍ**  
zobrazit galerii

## Kapitoly článku

Staňte se mistry Příkazové řádky

Staňte se mistry Win+R

Počítač s Windows můžete ovládat [pohybem očí](#), [pohybem ruky](#), [hlasovými příkazy](#), [šoupáním prstu po dotykovém displeji](#), popojížděním myšáka po stole a nakonec [zadáváním klávesových zkratk](#) a bušením do funkčních kláves. Tím zpravidla končí způsob ovládání počítače u drtivé většiny domácích uživatelů, přičemž ne jeden fanoušek Linuxu se jen ušklíbne, že mu k tomu všemu přeci stačí jedno jediné černé okénko s blikajícím kurzorem.



[Začněte skutečně využívat počítač.  
Programujte ho](#)

Troufám si tvrdit, že toto zažité klišé je zhruba stejně tak pravdivé jako rčení, že co Čech, to muzikant a podobná mytologie. Uživatelsky přívětivé linuxové distribuce dnes s trochou štěstí nevyžadují pokročilejší konfiguraci, a když na to přijde, běžný uživatel beztak bude pátrat dlouhé hodiny ve všemožných diskuzích, poněvadž tomu sám nerozumí. Pokročilý uživatel a systémový administrátor pak umí svůj operační systém skriptovat nehledě na to, jestli se jedná o linuxovou distribuci, nebo Windows.

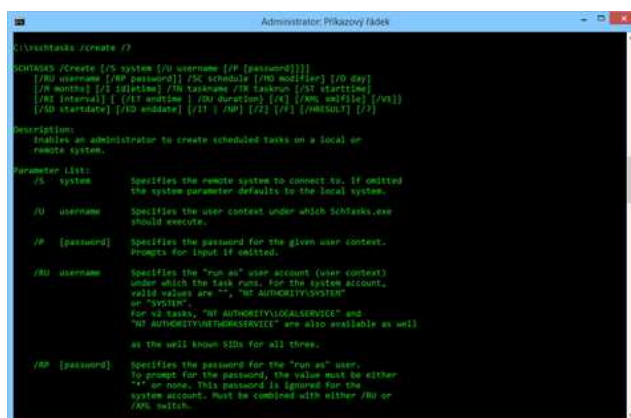
Zatímco pokročilý linuxový správce se zaštití znalostí [Bashe](#) a [Pythonu](#), ten ze světa Microsoftu se zase vytasí třeba znalostí [PowerShellu](#). Oba si nicméně mohou podat ruce.

## Referenční seznam příkazů pro textovou konzoli na Windows 8, 7 i XP najdete na stránkách Microsoft TechNet.

Přítomnost textové konzole – příkazové řádky ve Windows se však vyplatí připomenout i obvyklým počítačovým smrtelníkům, kteří jsou zvyklí na myš a hektické klepání po ikonkách. Stejně jako znalost elementárního programování je to totiž další silná zbraň, která dokáže za určitých okolností výrazně zrychlit práci s počítačem a dokonce i zpřehlednit jeho konfiguraci.

## Staňte se mistry CMD – příkazové řádky

Pokročilá a volitelná příkazová řádka Microsoftu jménem PowerShell toho dnes umí opravdu mnoho včetně slušného skriptování, nicméně na takové to domácí hraní bohatě stačí i klasický **CMD.exe**, tedy Command Line – česky Příkazový řádek.



```
Administrator: Příkazový řádek

C:\Users\user>schtasks /create /?

SCHTASKS /Create [/s system [/u username [/p password]]]
        [/ru username [/rp password]] /TN schedule [/M modifier] [/D day]
        [/M modifier] [/D day] /RU username /TR taskname [/ST starttime]
        [/RI interval] [/IT endtime] [/DU duration] [/A] [/XN noFile] [/V]
        [/SD startdate] [/ED enddate] [/DT] [/P] [/Z] [/F] [/MISMSI] [/F]

Description:
        Enables an administrator to create scheduled tasks on a local or
        remote system.

Parameter list:
        /s system           Specifies the remote system to connect to. If omitted
                           the system parameter defaults to the local system.

        /u username         Specifies the user context under which SchTasks.exe
                           should execute.

        /p [password]       Specifies the password for the given user context.
                           Prompts for input if omitted.

        /ru username        Specifies the "run as" user account (user context)
                           under which the task runs. For the system account,
                           valid values are "", "NT AUTHORITY\SYSTEM"
                           or "SYSTEM".
                           For all tasks, "NT AUTHORITY\LOCALSERVICE" and
                           "NT AUTHORITY\NETWORKSERVICE" are also available as well
                           as the well known SIDs for all three.

        /rp [password]      Specifies the password for the "run as" user.
                           To prompt for the password, the value must be either
                           "*" or none. This password is ignored for the
                           system account. Must be combined with either /ru or
                           /XN switch.
```

Žádné stínečky, efektičky a další zbytečnosti. To je příkazová řádka.

Historicky vychází z DOSu a mnozí mu tak dodnes říkají, nicméně fakticky s ním nemá nic společného a pouze přebírá některé jeho příkazy a emuluje některé nástroje (většinou jen na 32bitových verzích Windows).

Stejně jako na Linuxu i na Windows nabízí mnoho systémových programů přístup jak skrze grafickou verzi, tak tu textovou a skriptovatelnou v příkazové řádce. Přiznám se, že ta druhá jmenovaná mi občas připadne jednodušší a přehlednější.

## Chci automatické spouštění Outlooku

Krásným příkladem je třeba plánování všemožných úloh, které má počítač provést třeba každý den ve tři odpoledne. Tučňáci mají svůj **CRON**, uživatelé Windows ale mají stejně vybavený **SCHTASKS**.

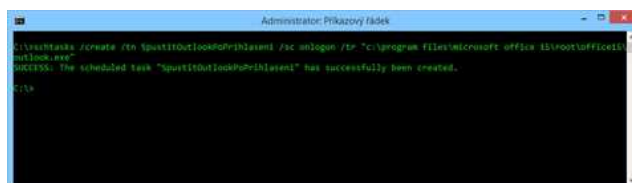
Program schtasks.exe je textové prostředí pro grafický **Plánovač úloh**. Některé složitější úlohy mohou být pro začátečníka snazší právě v grafickém prostředí, textový program ale zase vytvoří vše naráz jediným příkazem. Výsledkem pak může být úloha, která bude třeba každý týden v úterý ve tři odpoledne spouštět nějaký program, či provádět jiné úkony.

**Příklad:** Při příchodu do práce zapomínám spouštět Outlook, takže si naplánuji, aby se mi po přihlášení spustil automaticky. Do příkazové řádky tedy napíšu:

```
schtasks /create /tn SpustitOutlookPoPrijhlaseeni /sc onlogon /tr
"c:\program files\microsoft office 15\root\office15\outlook.exe"
```

**Použité parametry:**

**/CREATE** – budu vytvářet novou úlohu  
**/TN** SpustitOutlookPoPrijhlaseeni – **T**ask **N**ame, tedy název úlohy  
**/SC onlogon** – **S**chedule, tedy spouštěč úlohy; v tomto případě okamžik přihlášení  
**/TR** „c:\...“ – **T**ask **R**un, tedy program, který má úloha spustit



To je všechno. Jeden příkaz, jedno potvrzení. Tečka. Úloha je připravená.

Nebýt ošklivě dlouhé cesty k programu Outlooku, byl by to opravdu jednoduchý příkaz, který na jednom řádku vytvořil úlohu, která se jmenuje SpustitOutlookPoPrijhlaseeni a bude se spouštět, jakmile se uživatel přihlásí do Windows.

Podobných spouští jako **ONLOGON** může být samozřejmě celá hromada počínaje konkrétním časem, periodickým opakováním (každou x. minutu, hodinu, den, týden, měsíc aj.), případně to může být nějaká změna stavu (po spuštění PC, přihlášení k účtu), nebo nakonec nějaká jiná blíže specifikovaná událost. Můžete také připojit jméno a heslo uživatele, pod kterým má být úloha provedena, pakliže se netýká přímo vašeho účtu.

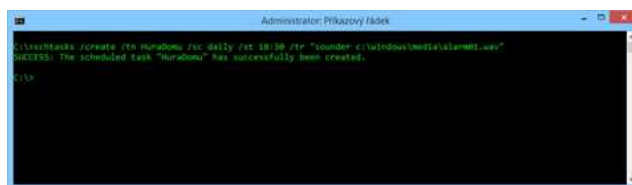
## Chci budík, který mě upozorní na to, že padla

Takže ještě jednou. Dejme tomu, že si chci vytvořit úlohu, která každý den v 18:30 přehraje nějaký alarm jako upozornění, že je čas vyrazit z kanceláře domů, respektive na pivo. V tomto případě si pomohu jednoduchým textovým přehrávačem [Sounder](#). Vytvořím tedy takovouto úlohu:

```
schtasks /create /tn HuraDomu /sc daily /st 18:30 /tr "sounder c:\windows
\media\alarm01.wav"
```

**Použil jsem nový parametr:**

**/ST** – **S**tart **T**ime, tedy čas začátku ve formátu HH:MM



A je to, další jednoduchá úloha čeká na vyřízení

Úloha se tedy bude jmenovat HuraDomu, bude se spouštět každý den v 18:30 a bude volat program sounder.exe, který má jako parametr cestu ke zvukovému souboru WAV z bohaté knihovny přímo v systému. Program sounder.exe jsem umístil do složky Windows, případně ho umístím do jiného adresáře a v konfiguraci (Win+Pause/Break → Upřesnit nastavení systému → Proměnné prostředí) jej zařadím do systémové proměnné Path, abych pokaždé nemusel zadávat plnou cestu.

## Chci jednou týdně automaticky mazat Koš

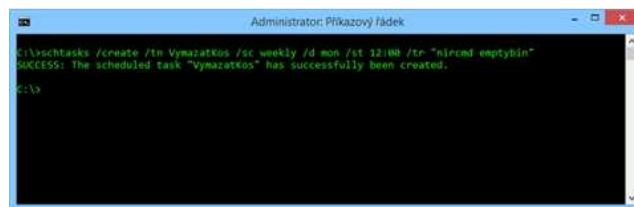
Do třetice ještě jeden příklad. Napišu si úlohu, která mi bude každý týden v pondělí odpoledne promazávat systémový Koš. Hromadu podobných užitečných kousků umí třeba textový prográmk [Nircmd](#). Stačí ho prostě zavolat s adekvátním parametrem.

Časová úloha by tedy mohla vypadat třeba takto:

```
schtasks /create /tn VymazatKos /sc weekly /d mon /st 12:00 /tr "nircmd emptybin"
```

**Použil jsem nový parametr:**

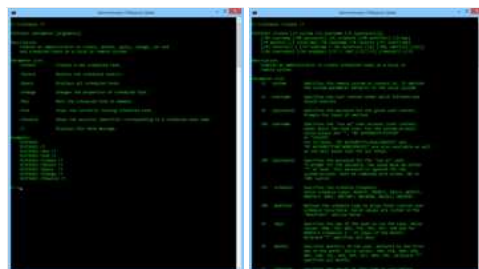
**/D mon** – Day, tedy den v týdnu, ve kterém se má úloha provést (mon = pondělí)



A od tohoto okamžiku se mi každé pondělí v poledne v tichosti vymaže Koš

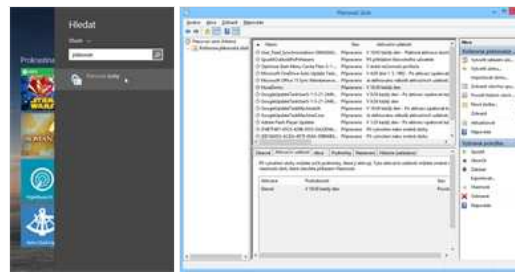
## Jde to i graficky, ale kdo se v tom má vyznat?

Práce s textovou verzí plánovače úloh vám může na první pohled připadat složitá, ale je to opravdu jednoduché. Pro nápovědu stačí napsat **schtasks /create /?** a zobrazí se vám manuál se všemi parametry pro vytváření úkolů. Pokud chcete znát i další operace plánovače včetně úprav a mazání existujících úloh, napište **schtasks /?**.



Základní nápověda textového schtasks a modulu pro tvorbu nových úloh

Toto všechno samozřejmě můžete provést i v grafickém plánovači (ve Windows napište do nabídky Start/Metra **Plánovat úlohu**), ale budete se muset prokousat nepříliš přehledným programem. Jako případný programátor pak přijdete o možnost skriptování – automatizace. Vždyť takový příkaz můžete uložit do běžného textového souboru, dát mu příponu BAT a máte spustitelný skript příkazové řádky, na který pak už stačí pouze poklepat a vše se zpracuje.

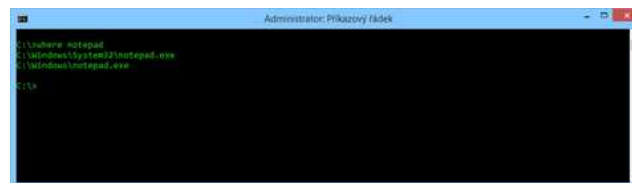


Jistě, všechno můžete nastavit i v grafické podobě plánovače, ale nejprve jej budete hodinu studovat, jak se vůbec ovládá...

## Hledáme soubor pomocí WHERE

Příkazová řádka Windows disponuje celou plejádou textových nástrojů. Chcete najít soubor? Můžete použít příkaz **WHERE**.

Nejjednodušší použití vypadá takto: **where notepad**.

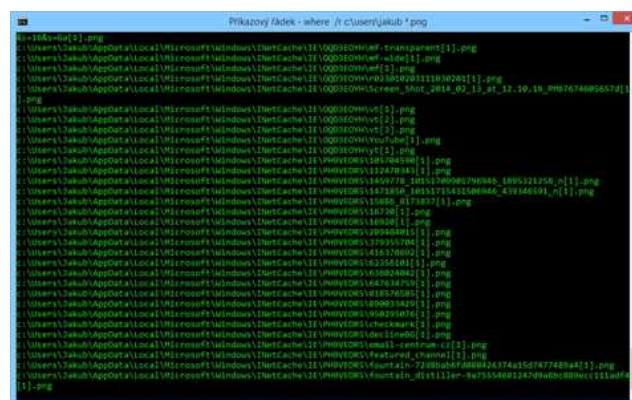


Kdepak asi je?

Program WHERE v tuto chvíli neví, kde má vlastně hledat, použije tedy předdefinované adresáře v systémové proměnné Path. Notepad.exe je v jednom z nich, takže se dozvíte, kde je ve Windows vlastně uložený. Poznámkový blok.

Pokud naopak chcete najít všechny obrázky PNG ve vašem uživatelském profilu, napište třeba:

**where /r c:\users\jakub \*.png**



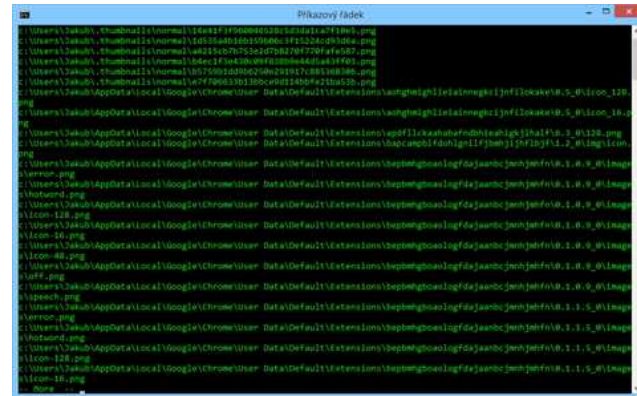
A sakra, toto bude na dlouho. Spuštěný program naštěstí můžet zastavit příkazem CTRL+C

Program rázem začne rekurzivně (parametr **/R**) procházet všechny vnořené adresáře v uživatelské složce Users\Jakub a soubor po souboru bude zjišťovat, jestli obsahuje výraz \*.png.

## Roura |

Při takto velkém rozsahu to může chvíli trvat, čili na konec příkazu můžete připojit ještě rouru **MORE**:

where /r c:\users\jakub \*.png | more



**Hurá, roura zaplnila obrazovku a čeká, až stisknu mezerník**

Roura přesměřuje standardní výstup programu **WHERE** do standardního vstupu programu **MORE**, jehož účelem je výstup stránkovat. Příkazová řádka se tedy celá zaplní a výpis se zastaví. Stačí stisknout (nebo držet) posuvník a budete se tedy výsledky postupně po stránkách posouvat dál.

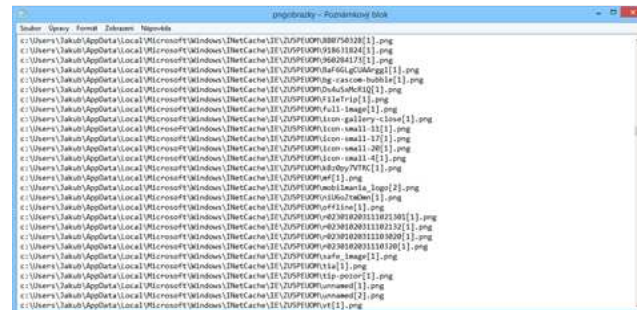
## Přesměrování > a >>

Co když ale budete chtít tento výstup uložit do textového souboru? V takovém případě poslouží parametry `> a >>`. Ten první textový výstup uloží do souboru, který přepíše, ten druhý připojí výstup na konec souboru, pakliže v něm už něco je.

Takže když napíšete:

**where /r c:\users\jakub \*.png > c:\users\jakub\pngobrazky.txt**

v příkazové řádce se nic vypisovat nebude, ale po ukončení programu celý seznam najdete v textovém souboru pngobrazky.txt.



Výstup z příkazové řádky jsem přeměroval do běžného textového souboru

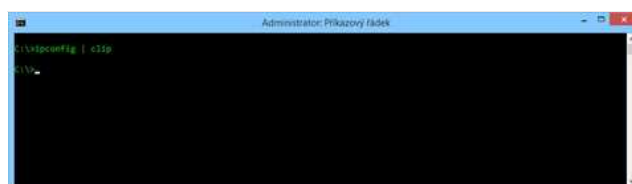
Kopírujeme do schránky



U menších výstupů se hodí text přesměrovat do schránky Windows. To oceníte v případech, kdy chcete s výstupem dále pracovat – třeba odeslat výstup programu **IPCONFIG** technikovi, když vás zlobí internetové připojení.

Pro přesměrování výstupu do schránky opět použijte rouru a program **CLIP**:

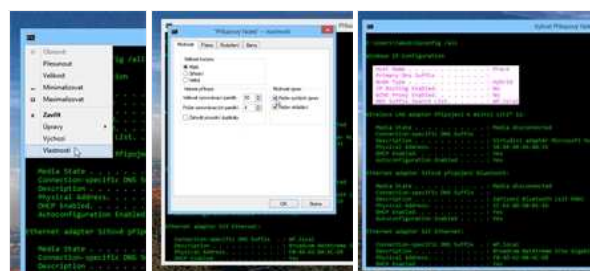
**ipconfig | clip**



Do příkazové řádky se nic nevypsalo, neboť se to vypsalo do schránky

Tak a teď si schválně v nějakém textovém editoru klepněte na CTRL+V.

Pokud byste chtěli výstup v příkazové řádce kopírovat tradiční cestou označením textu myší, v dialogu **Vlastnosti** klepněte na kartu **Možnosti** a zaškrtněte **Režim rychlých úprav**.



Označení textového výstupu myší pro snadné zkopírování výřezu

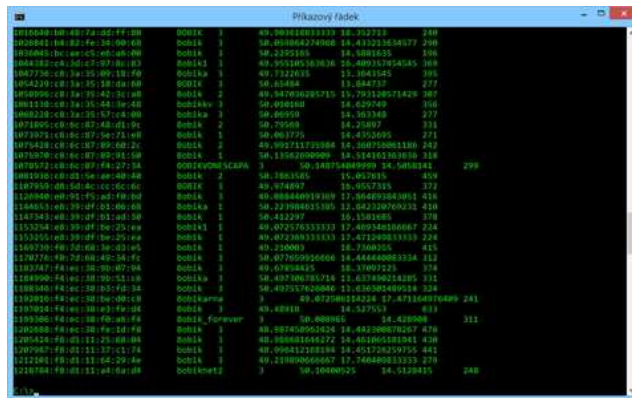
## Hledáme v textu pomocí FINDSTR a FIND

Windows disponuje desítkami a stovkami vestavěných textových programů a všemožných příkazů. Na konec tedy snad ještě hledání. Dejme tomu, že budete mít obrovský textový soubor. Třeba databázi českých Wi-Fi sítí z projektu Wifileaks, jejíž loňská verze je [volně ke stažení z této adresy](#). Jedná se o rozměrný textový soubor o velikosti bezmála 72 MB, přičemž na každém řádku je záznam o jedné Wi-Fi síti. Celkem jich je v seznamu více než milion.

Jak je hledat? Otevírat takto rozměrný soubor v Poznámkovém bloku může být krajně nepraktické (pomůže rychlý [Lister](#) od Ghislera), naštěstí tu ale máme další textový program **FINDSTR**.

Dejme tomu, že budete chtít v databázi vyhledat všechny Wi-Fi sítě, které mají v názvu slovo **Bobik**. Pro FINDSTR je to hračka, stačí zadat:

**findstr /in "bobik" wifileaks.tsv**

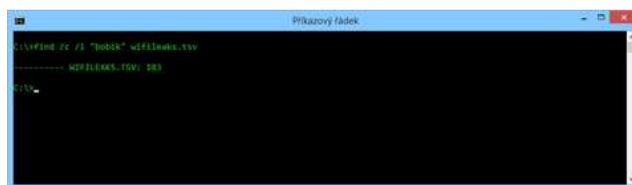


**To je ale bobíků!**

Do textového okénka se prakticky okamžitě vypíšou všechny řádky, které obsahují řetězec „bobik“. Použil jsem dva parametry **I** a **N**. Parametrem I dávám programu na vědomí, aby se neohlížel na velká a malá písmena (vyhledá se Bobik i bobik) a parametr N zase na začátku každého vypsaného záznamu napíše číslo řádku ve zdrojovém souboru pro případné snazší dohledání.

No, a jelikož už znáte funkci přesměrování a roury, můžete na konec opět připojit třeba **|MORE** nebo **> nalezenibobici.txt**.

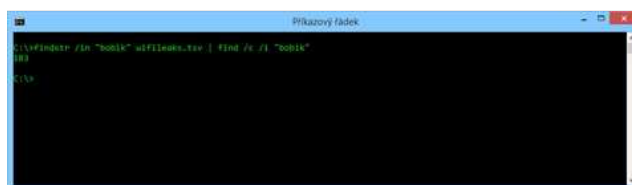
Z výpisu se bohužel nedozvím, kolik oněch sítí vlastně je a FINDSTR samotný žádný takový parametr nemá. Mohu ale použít starší vyhledávací program **FIND** a jeho parametry **/C** – počet a **/I** – nerozlišovat malá a velká písmena. Hledání je nicméně oproti FINDSTR výrazně pomalejší; tedy alespoň v takto velickém souboru.



**Zjištění počtu bobíků. Je jich 183, ale trvalo to několik dlouhých sekund.**

Počítání programu FIND je natolik pomalé, že se vyplatí opět roura. Nejprve vyhledám seznam všech řádků se slovem bobik pomocí FINDSTR a tyto výsledky přepořžu rourou do programu FIND, kde řádky spočítám. Je to sice zdvojené vyhledávání, ale i tak to bude mnohem rychlejší:

```
findstr /in "bobik" wifileaks.tsv | find /c /i "bobik"
```



### Spočítání bobíků ještě jednou a mnohem rychleji - prakticky okamžitě

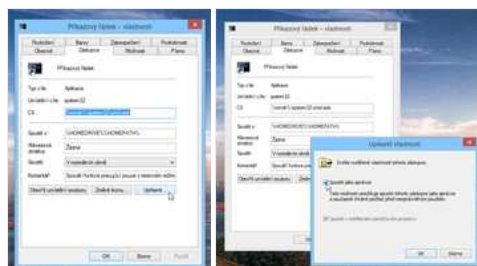
## Admin

Mnoho systémových operácií v príkazovej řádce provedete pouze jako administrátor. Windows bohužel nezná nic jako příkaz **SU** a **SUDO** ze světa Linuxu, samotný program CMD tedy už musíte spustit s patřičnými právy.



Buď tak učiníte přes kontextové menu nad zástupcem/programem a volbu **Spustit jako správce**, nebo do nabídky Start/Metra napíšete jednoduše **CMD**, stisknete **CTRL+SHIFT+ENTER** a příkazová řádka se rovnou spustí s právy administrátora.

Pokud byste ji chtěli jako správce spouštět ve výchozím stavu, je třeba upravit zástupce. Otevřete jeho dialog **Vlastnosti** a na kartě **Zástupce** klepněte na **Upřesnit**. V novém dialogu pak můžete zaškrtnout, aby se vybraný program vždy spouštěl s administrátorskými právy.

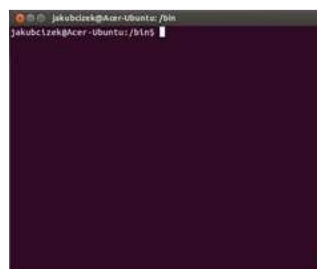


Nastavení zástupce, aby příkazovou řádku pokaždé spouštěl s právy administrátora

Jen připomenu, že k nastavení zástupce libovolného programu v Metru se dostanete přes kontextovou nabídku a volbu **Otevřít umístění souboru**.

## Vtípek na konec: CMD jako linuxový terminál

Zatímco většina linuxových emulátorů terminálu uvozuje blikající kurzor textem **uživatel@počítač: aktuální adresář\$**, příkazová řádka Windows už od dob DOSu uvozuje vstup textem **Aktuální adresář>**. Typicky tedy třeba **C:\>**.

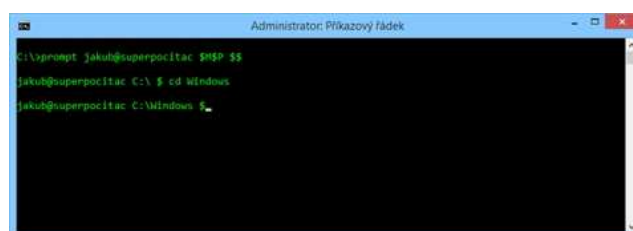


Textová konzole na Ubuntu

Příkazová řádka je nicméně tvárná, čili si můžete nastavit naprosto libovolný uvozovací text a to pomocí příkazu **PROMPT** (nápoředu s možnými makry najdete tradičně na prompt /?).

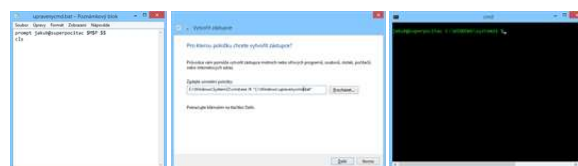
Dejme tedy tomu, že si budete chtít nasimulovat třeba onen terminál z Linuxu. Pak byste mohli použít něco podobného:

**prompt jakub@superpocitac: \$M\$P \$\$**



A už mám „skoro“ Linux

Toto nastavení není trvalé a jakmile příkazový řádek zavřete, nastavení se smaže. Stačí ale příkaz společně s vyčištěním výstupu pomocí **CLS** uložit do souboru BAT a vytvořit k němu zástupce, který bude směřovat třeba na **cmd.exe /k "c:\windows\upravenycmd.bat"**.



Upravená příkazová řádka s „linuxovým“ úvodním textem

## Svět za zrcadlem

Pamětníci DOSu a zkušení uživatelé a správci Windows se teď možná jen posměvačně ušklíbli a mají na jazyku výtku, že toto všechno přeci všichni vědí – vždyť DOS tu používala celá jedna počítačová generace. Jenže to bylo kdysi dávno v minulém století a znalost textového prostředí pomalu mizí. Troufám si tvrdit, že většina současných domácích uživatelů Windows ani netuší, že se v něm něco podobného vůbec nachází.

Je to docela škoda, jednoduchý textový program může být totiž často snadnější na pochopení a používání než složitá grafická aplikace, ve které marně hledáte to správné tlačítko a položku v menu.

Textové prostředí prý není sexy, ale stejně jako v mém předchozím článku [Začněte skutečně využívat počítač. Programujte ho](#) platí, že kdo jej alespoň v základu pochopí a dokáže ocenit jeho přímočarost a rychlost, stane se opravdovým pánem svého PC.

ETARGET odkazy

Kapitoly článku

Diskuze (120) | Další >

Staňte se mistry Příkazové řádky

[Staňte se mistry Win+R](#)

**Témata článku:** Software, Windows, Operační systémy, Programování, Celý seznam, Textový nástroj, Blikající kurzor, Řád, Nový parametr, Textové prostředí, Tn, Textová konzole, Ask, Su, Guide, Strong, Bashe, Bobík, Automatické spouštění, Textový terminál, Podobné parametry, Shell, Wine, Jednoduchý příkaz, Dialog

**Michio Kaku: „Mimozemšťané nás kontaktují v tomto století!“**

**Podívejte se na snímek jediného atomu „zamrzlého“ v pohybu a další úžasné obrázky**

**Studie odhalila, proč se afričtí sloni děsí kvadrokoptér**

**Aerosoly z elektronických cigaret obsahují**

**toxické těžké kovy**

**Jak zachraňovat životy na bojišti?  
Zpomalením biologického času!**

**Peklo opět zamrzlo: V Microsoft Storu je  
hackerský Kali Linux**

**Americké školy okupuje Chrome OS, ty  
světové Windows**

**Bitcoin má problém a letí dolů: velká burza  
Binance se možná stala obětí hackerů**

**Amazon Echo straší své uživatele. Bez  
varování jim v noci pouští děsivý smích**

**„Havarované UFO“ na Google Earth je ve  
skutečnosti přírodní fenomén**

---

**živé firmy.cz**

**Největší nezávislý katalog českých firem**

Hardware

Počítačové sítě

Školení IT

Internet marketing

Připojení k internetu

Webdesign

IT poradenství

Software

Webhosting

Najděte firmu nebo službu

**Hledat**

[Přejít na Živéfirmy.cz >](#)



**Spekulace: K vývojářům třetí strany už míří dev kity konzole PS5**



**Toyota přišla na to, co dělat se starými bateriemi hybridů. Lepší to být nemohlo!**



**V akci s EOSem 77D a objektivy 24 mm a 35 mm F2,8 macro**



**FOTO: Trvalky, i zboží na jedno použití. 41 sexy fotbalových (ex)partnerek**



**Je to trapné, nechutné a vulgární, ale stejně vám to ukážeme. Aplikace Lickster učí lízat**



**Mapa Kingdom Come: Deliverance je krásným příkladem středověké historie**



**Nové Wiko je hezčí než iPhone X. Běží na Androidu a bude stát třetinu**



**S aplikací Halide budete mít dokonalou mobilní fotografii na dosah ruky**



**Český miliardář pokořil rekord. S Bugatti Veyron jel 402 km/h v běžném provozu!**



**Další značka skončuje s turbodiesely. Nejenom kvůli vysokým nákladům**

Doporučuje ContentExchange 