

# **World of Mirra 3server tutorialy**

kopírování jen se souhlasem autora a jen pro účely 3serveru.

# Obsah

World of Mirra 3server tutorialy.....	1
ÚVOD.....	2
Vysvětlivky a definice pro správné chápání tutorialů.....	2
Definice serverů .....	3
LIVE server .....	3
DEVS ONLY nebo Testovací server .....	3
Chudák .....	3
Realmd a authserver = autentizační server.....	3
Lekce 1.....	4
Přihlášení.....	4
Jak se na server přihlásit pomocí Putty přes windows.....	4
Jak se na server přihlásit pomocí Putty přes linux a mac.....	5
Lekce 2.....	6
Spuštění serveru, Vypnutí serveru, restarter.....	6
Krátký úvod .....	6
Spuštění serveru .....	6
Spuštění serveru na chvíli pro dočasné účely.....	6
Spuštění serveru na trvalo.....	6
Ověření, že server jede.....	6
Ize to taky ověřit příkazem pgrep worldserver .....	7
Jak server vypnout?.....	7
Slušně.....	7
Neslušně/drže.....	7
Restart World serveru.....	8
Neslušně, automatizovaně (neodbugovaně).....	8
Lekce 3 .....	8
Nastavení serveru, logy serveru.....	8
3 možnosti, jak zeditovat configurační soubory.....	8
Lekce 4 .....	10
Git, jak stáhnout vlastní zdrojáky a skompilovat si je podle návodu na Trinity core 2.....	10
Pro další zájem odkaz na zdroje v češtině a angličtině s komentářem.....	14
Lekce 5 .....	14
Další osvěžení znalostí linuxu.....	14
WINSCP.....	16

# ÚVOD

## ***Vysvětlivky a definice pro správné chápání tutoriálů***

Syntaxe příkazů bude kurzívou takto : *příklad*

Syntaxe příkazů bude kurzívou takto : příklad

<tabulátor> nebo <tab> znamená 1x stisknutí tabulátoru (tlačítka pod ;° a nad caps lockem)

Pojmy jméno a heslo, login a heslo, login a pass atd.. vyjadřují, když hovoříme o administraci serveru vždy to stejné a to vaše přihlašovací údaje NIKOLIV DO HRY, ale takové o jaké jste si řekli, případně takové, jaké vám byly přiděleny (přihlašovací údaje).

Pojem distribuce linuxu znamená druh linuxu. Například Fedora, Ubuntu, Debian, Suse, Open Suse, Redhat, Centos, mandriva, Mint, Arch, DSL linux.

Zkratka RHEL znamená Red Hat Enterprise Linux

TBC znamená The Burning Crusade.

Wotlk znamená Wraith of the Lich king.

Zkratka TC, TC2 znamená Trinity Core, Trinity-Core nebo TrinityCore2. Jedná se v podstatě o stejný projekt s tím rozdílem, že Trinity-Core existovalo pro datadisk TBC a Trinity-Core 2 existuje pro datadisk WoTLK.

Konfigurák znamená konfigurační soubor, zpravidla s příponou .conf .

Zdroják znamená textový soubor obsahující zdrojový kód.

## **Definice serverů**

### ***LIVE server***

Hlavní server pro World of Warcraft pro ostrý provoz, na kterém hraje nejvíce hráčů.

### ***DEVS ONLY nebo Testovací server***

Testovací server pro chystání nových revizí a zkoušení nových updatů a patchů serveru.

### ***Chudák***

Testovací server určený pro likvidaci Developery začínajícími a učícími se se serverem, či pro Gmka, která si chtějí zkusit příkaz, který by byl nebezpečný na hlavním LIVE serveru, nebo pro začínající SQL fixery. Nejedná se v případě chudáka o žádný ostrý provoz, když někdo něco zničí, musí to sice opravit, ale neohroží to provoz hlavního LIVE serveru, takže pro zkoušení, zaučování ideální server.

### ***Realmd a authserver = autentizační server***

Pokud běží worldserver, ale neběží authserver, hráči ve hře o ničem nevědí a hrají vesele dál, ale hráči kteří se chtějí na server připojit mají smůlu a na server se nedostanou.

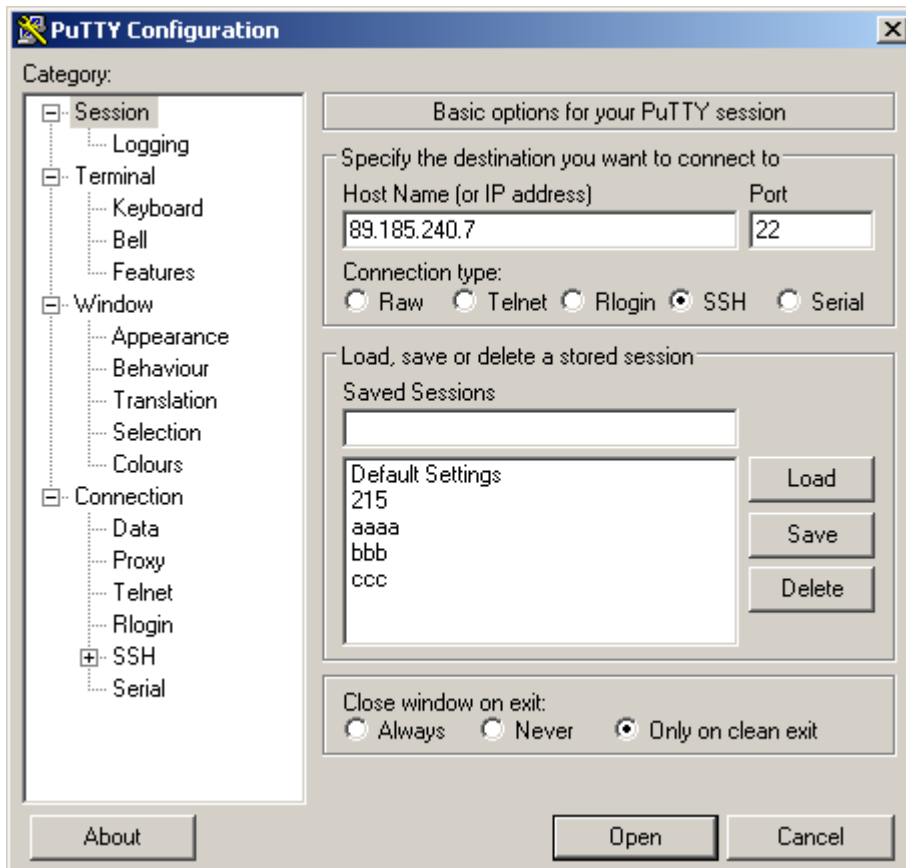
# Lekce 1

## Přihlášení

- I) Jak se na server přihlásit pomocí Putty přes windows (BEZ RSA šifrování)
- II) Jak se na server přihlásit pomocí Putty přes linux a mac (BEZ RSA šifrování)

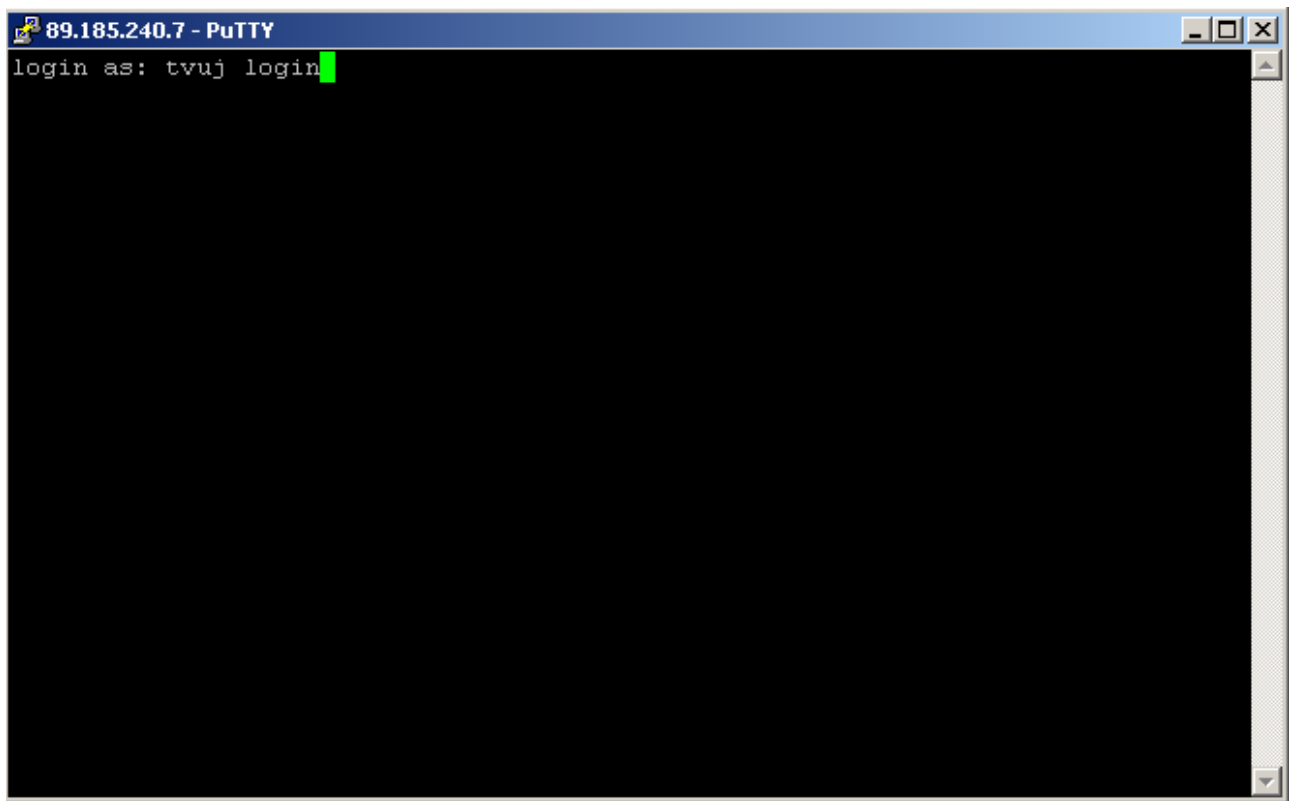
### Jak se na server přihlásit pomocí Putty přes windows

- 1) Stáhni putty zde <http://www.slunecnice.cz/sw/putty/>



Obrázek č. 1 : login přes putty

- 2) Vyplníš v HOST name adresu serveru. Chudák je 89.185.240.214 , wow server je 89.185.240.7 a test server „DEVS ONLY“ je 89.185.240.247. Hlavní stroj je 3server.net nebo 89.185.240.6.
- 3) Pokud se připojuješ bez RSA certifikátu, tak klikneš open a ....
- 4) Zadáš jméno a heslo a přihlíšíš se a jsi tam. Po přihlášení, pokud se jedná o ubuntu distribuce, zadáš `sudo su` a znovu se tě to zeptá na stejné jméno a heslo



Obrázek č. 2 : login přes putty na terminálu

## **Jak se na server přihlásit pomocí Putty přes linux a mac**

Otevřete terminál.

A napište ssh přihlašovacíJméno@ipAdresaServeru příklad :

ssh [franta@89.185.240.214](mailto:franta@89.185.240.214) dáte enter , zeptá se vás to na jméno a heslo a jste tam.

Pokud nemáte ssh nainstalovaný, tak dejte na ubuntu a debianích distribucích apt-get install ssh nebo sshd . Na Centosích, fedořích a takovýchto redhatích distribucích by to mělo být standardně nainstalováno. Kdyžtak yum install ssh .

## Lekce 2

### ***Spuštění serveru, Vypnutí serveru, restarter***

#### **Krátký úvod**

Trinity core a další řada serverů jako je mangos mají přihlašovací server a herní server. Někdo tomu říká auth a world, nebo login a world, nebo login a core, realmd a world, login a game atd... Na hlavním stroji 3server.net nebo 89.185.240.6 nastavujeme JENOM authserver neboli realmd. Představte si to jako rozcestí, na které dojde hráč a u toho rozcestí je značka a další cesty a hráč se rozhodne, jestli teda půjde na Live server, nebo na Devs only server (pokud má pravomoce větší alespoň plvl 2) anebo na chudáka (pokud má udělené pravomoce plvl 5 na chudákovi). Na dalších linuxových strojích a to .7 kde je Live, nebo .247 kde je Devs only, anebo na .214 kde je chudák už pouštíme JENOM WORLDSERVER. To je taky důvod, proč je realmlist hráčů stále stejný a nezávisí vůbec na tom, kde funguje herní server.

#### ***Spuštění serveru***

V podstatě všechny naše wow servery nezávisle na každém serveru jsou umístěny v :  
/trinity/trinity/

Konfiguráky trinity jsou vždy v /trinity/trinity/etc/

Restarter je vždy v /trinity/trinity/restart.sh na některých serverech jsem udělal navíc /etc/init.d/authrestarter, který přijme parametr start, stop, restart, status. Více dole.

#### **Spuštění serveru na chvíli pro dočasné účely**

Pokud chcete spustit server JENOM pro účely, jestli se to dostane přes správné přístupy do databáze, tak zadáte do terminálu buď :

a) `./trinity/trinity/worldserver`

anebo

b) `cd /trinity/trinity` čímž si zajedeme do adresáře, kde chceme spustit worldserver (pro doplňování syntaxe si můžete hodně pomáhat tabulátorem, napíšete třeba jen trini zmáčknete tabulátor a jestli to děláte správně, tak se vám to automaticky doplní, nebo nabídnou další možnosti.)

A potom stačí jen napsat buď `./worldserver` nebo `worldserver` (v závislosti na distribuci linuxu).

Ukončíte CTRL + C. (to si vždy jenom necháte naběhnout, jestli to fakt je schopno naběhnout asi prvních 5 vteřin, že vám to nezahavuje kvůli špatně nastaveným údajům do mysql.

#### **Spuštění serveru na trvalo**

Pokud chcete pustit opravdu na trvalo, tak spustíte server přes restarter a to tak, že odkloníte standardní vstup do souboru takto :

```
nohup /trinity/trinity/restart.sh &
```

nohup znamená spustit a běžet i po odhlášení uživatele, který proces spustil

& znamená spustit na pozadí, takže lze pracovat s příkazovou řádkou dál.

#### **Ověření, že server jede**

Buď si dáte příkaz `top`, který vám vypíše nejžravější proces a protože je WoW server obvykle nejžravější, tak vám ho to vypíše na prvním místě nahoře. Ukončíte q.

Info z ubuntu wikiny :

Příkaz **top** ("top" - vršek, špička) zobrazuje informace o linuxovém systému, běžící procesy a systémové prostředky, včetně CPU, RAM & swap a počet právě běžících úkolů. Pro skončení příkazu **top** stiskněte "q".<sup>1</sup>

### **Ize to taky ověřit příkazem pgrep worldserver**

Pokud příkaz vrátí výsledek v podobě nějakého čísla třeba 4542 nebo 18754 (to číslo je náhodné) tak máte jistotu, že worldserver běží, protože to co vám to vypíše je číslo jeho procesu. To stejné lze udělat s restartérem a to příkazem `pgrep restart.sh`. Pokud restarter běží, tak vám to po zadání `pgrep restart.sh` vypíše číslo jeho procesu. Pokud u žádného z příkazu nevypíše nic, tak máme jistotu, že daný proces neběží. Pokud vám příkaz `pgrep NázevProcesu` vypíše více čísel PID, tak jste buď spustili nějakou aplikaci či process 2x, tak se vám PID vypíše víc. Pokud by jste ale třeba dali jen `pgrep a`, tak vám to vypíše všechny procesy, které začínají na a, takže pozor na to, doporučuji psát minimálně 95% názvů procesů, jinak killnete kromě serveru i něco, co killnout nechcete o čemž hovoří další podkapitolka.

## **Jak server vypnout?**

### **Slušně**

Server lze vypnout **slušným** způsobem. A to tak, že dáme v příkazové řádce `pgrep restart.sh`. To nám vypíše PID (neboli process ID = číslo procesu) a pak stačí jen v dalším příkaze napsat `kill čísloProcesu` které nám vyplivnul `pgrep restart.sh`. Anebo to lze udělat rovnou jedním příkazem a to příkazem :

```
kill $(pgrep restart.sh)
```

Příkaz `kill` bez optávání zabije process (vypne ho násilně) a vše co je v `$( )` je výsledek příkazu, který poslouží jako taková dočasná proměnná. Vypne to všechny instance stejného procesu, takže pokud by se stalo, že se pustil `restart.sh` vícekrát, zabije to všechny procesy s názvem `restart.sh`. Z PHP všímavější jedinci vědí, že proměnné se značí \$ dolarem v linuxu i v bashi, je to tedy v podstatě velmi podobné.

Tím, že jsme zabili `restart.sh` jsme se ujistili, že po vypnutí serveru už znovu server nenaběhne. No more Jesus Christ again .-) Všem věřícím se tímto omlouvám za předchozí větu.

Potom už stačí jenom někomu z GM teamu říct, aby se přihlásil do hry a vypnul server přes příkaz. Zpravidla v době psaní článku to bylo příkazem ve WoW na TC2 [server shutdown Počet vteřin](#), takže [server shutdown 30](#) (třeba). Pokud se jedná o testovací server, můžete dát klidně 5 nebo i 3 vteřiny, klidně i 1 vteřinu. Pokud se jedná o LIVE (hlavní) server, ať se GM poptá, jestli neprobíhá zrovna nějaký raid, nebo PVP bitva, BG, arény apod...To v takovém případě musíte složitě vyjednávat s hráči a domluvit se třeba na 10 minutách. 10 minut = 600 vteřin pro méně zdatné. Hlavně se při vypínání serveru pro jakýkoliv druh údržby ujistěte, že jste killnuli `restarter.sh`, aby to už znovu nenaběhlo.

### **Neslušně/drže**

**Neslušně/drže** vypnout lze následujícím způsobem.

Napišeme : `kill $(pgrep restart.sh)` a pak `kill $(pgrep worldserver)`

To však ale moc nedoporučuji, protože se neuloží herní pozice za poslední cyklus ukládání, protože process defakto odstřelíme jako vrabce vzduchovkou na zahradě. Stejně tak jako to nerozchodí onen vrabec, to nerozchodí ani veškeré rutiny, které se volají při slušném vypínání serveru ze hry. Tohle zkrátka doporučuji dělat pouze na chudákovi, když worldserver potřebujeme rychle zabít, když dlouho nikdo nebyl na DEVS only, tak to lze killnout eventuelně taky, ale můžeme tím udělat více škody než užitku. Takže určitě neslušnou metodu NEDĚLAT na LIVE serveru na ostrém provozu a pokud možno, tak raději neslušnou metodu NEDĚLAT ani na dev's only, protože pokud děláme případnou chystanou novou revizi, tak by nám to mohlo něco poškodit, nebo alespoň „zblbnout“. Ale na chudákovi to takhle můžeme vypínat dle libosti, protože tady to nikomu z hlediska herní kvality nevaadí.

<sup>1</sup> Zdroj : <http://wiki.ubuntu.cz/Z%C3%A1kladn%C3%AD%20p%C5%99%C3%ADkazy>

## Restart World serveru

Když chcete jakýkoliv WoW server **restartovat**, tak stačí zkrátka ve hře napsat příkaz [server\\_shutdown](#) [PočetVteřin](#) a on se za několik sekund oficiálně vypne, ale protože na linuxovém serveru běží restart.sh, který se postará o to aby worldserver po 20 vteřinách automaticky nahodil, tak zase worldserver naběhne. Jak jednoduché, že? :)

Poznámka : (restart.sh každých 20 vteřin testuje, jestli existuje nějaký process s názvem worldserver pomocí příkazu pgrep worldserver. Nic víc v tom není. Je to skript na cca. 7 řádků.).

## Neslušně, automatizovaně (neodbugovaně)

Neslušný, ale rychlý automatizovaný způsob jak to nahodit, nebo vypnout. Není však ale ještě plně odladěný ( na chudáku už běží, na devs only ještě ne a na hlavním stroji 3server.net se nestará o world ale o authserver), proto je lepší umět všechny předchozí způsoby. Je příkaz :

(a pomáhejme si prosím tabulátorem)

```
/etc/int.d/authrestarter start nastartuje restart.sh a worldserver
```

```
/etc/int.d/authrestarter stop killne restart.sh i worldserver
```

```
/etc/int.d/authrestarter status
```

 Na Devs only a chudák vypíše Worldserver je: ONLINE nebo OFFLINE a pokud je ONLINE tak vypíše PID procesu. Na 3server.net vypisuje to stejné s realmd serverem neboli authserverem a odtud taky vychází tento název. Udělal jsem to pořádně na hlavním serveru a pak to jenom skopiroval na ostatní a přepsal jen vnitřek.

## Lekce 3

### Nastavení serveru, logy serveru

Protože máme naše Trinity servery umístěné v /trinity/trinity, konfiguračky jsou umístěny v /trinity/trinity/etc .

Na herních serverech nastavujeme jenom worldserver.conf a na authserveru na 3server.net nastavujeme jenom authserver.conf.

Ukázka worldserver.conf :

<http://code.google.com/p/trinitycore/source/browse/src/server/worldserver/worldserver.conf.dist?r=d98edb9bef03a646f571c17a095ae3339ed05871>

Nastavují se v něm věci od ratů, přes jazykové rozlišení až po specifikaci cest, kde jsou mapy, vmapy, limit maximálního počtu online hráčů, kam a co a jak logovat, jaké chaty povolit, zakázat, komu zpřístupnit, jestli povolit na accountech jen alianci či hordu, nebo umožnit hráčům na jednom accountu vytvářet jak alianci tak hordu zároveň, možnost jestli by si mohli hráči hordy s aliancí a naopak rozumět, inspekce talentů, některých oprávnění GM, jako příkaz Gm.visible, Gm.LoginState , autorbroadcast ( pozor tuším, že je to potřeba zakompilovat do jádra, to tam není standardně), nastavení battlegroundů, arén a vzdáleného přístupu(POZOR! Vzdálený přístup je důležitý kvůli kredit systému, takže nezakazovat a poptat se webmasterů pro jistotu, jestli je správné jméno a heslo a automaticky při každém nahození ozkoušet) a další takové podobné vychytávky a nastavení.

### 3 možnosti, jak zeditovat configurační soubory

Hgz

1) Buď si soubor stáhnete k sobě do počítače pomocí programu WINSCP, kde je stejný login jako na přes putty nebo ssh. Editnete ho, a pak nahrejete zpátky zpět a nastavíte pravomoce viz kapitola č. 5 příkaz chmod.

2) Nebo použijete program mc (pokud na serveru není tak dáte apt-get install mc -y). mc je super věc. Je to souborový manažer připomínající M602 nebo norton commander ze starých dob 486tek, ale funguje hezky v příkazové řádce, je přehledný, je jednoduchý a umožňuje pohyb v souborovém systému linuxu velice jednoduše a efektivně. Má v sobě i FTP klienta. Funguje tam i klikání myší, takže na jakoukoliv nabídku kliknete, tak se vám



taky rozklikne. Více o mc v kapitole č.5. Zde si popíšeme jenom postup. Pomocí mc můžete přes F3 prohlížet obsah bez editace, anebo pomocí F4 zapisovat a editovat. V podstatě po zmáčknutí F4 se vám rozjede comand line textový editor nano, který má menu v češtině dole, takže je to intuitivní. Více v kapitole č.5.

3) Textový editor vim (opět kapitola č.5). V podstatě pomocí shift + i můžete zapisovat. Pak dáte ESC a napíšete :wq (pro uložení a ukončení) pokud nechcete nic ukládat a jen vyjet, tak dáte :q! Anebo jen :w pro uložení, ale nevyjždění. Píšu tu hlavně :q protože hodně začátečníků zkouší ctrl + c a to vám v tomhle případě fungovat nebude. Takže :q pro exit z vimu .

Ve worldserver.conf se nachází vždy kompletní nastavení worldserveru, jsou tam komentáře, takže vůbec nemusí mít někdo strach, že by nevěděl, co zadaná hodnota znamená. Je tam vždycky v komentářích napsáno, co je defaultní hodnota atd.. V tomto configu se nastavují přístupy do databáze, takže je to fakt klíčové nastavení serveru, bez kterého by server nenaběhl. Jen taková rada, přihlašovací údaje jsou nabídnované na ip adresu, to znamená , že ikdyby jste chtěli přihlašovací údaje ukrást, na jiném počítači s jinou ip adresou než ip adresou příslušného serveru se zkrátka do databáze nepřihlásíte, což je velice účinné bezpečnostní opatření, aby pokud už někdo ví heslo a jméno, tak se nepřipojí, pokud nemá danou ip adresu, která je nastavena u vašeho mysql účtu v mysql serveru. Protože máme traffik jen 200gb měsíčně maximálně, je nutné nastavit maximální kompresi packetů a to na hodnotu 9 z 9ti.

Proto jsem se rozhodl raději z konfiguráků tuto část vytáhnout abych ji názorně ukázal :

```
# Compression
#     Compression level for update packages sent to client (1..9)
#     Default: 1 (speed)
#     9 (best compression)
```

My musíme mít nastaveno :

```
Compression = 9
```

Tím, že dáme velkou kompresi nejenže šetříme trafik dat serveru i pro jiné projekty, ale taky to umožňuje v podstatě na 100mbitové lince mít i větší množství klientů, protože se nemusí posílat tolik dat. Je to i ekologické, protože nenutíme posílat celou internetovou infrastukturu tolik dat, ale v podstatě jenom jejich zlomek. Proto na tohle prosím kladme velký důraz a nastavujme vždy největší a nejlepší kompresi packetů.

Pokud někomu budete posílat worldserver.conf či authserver.conf vždy se ujistěte, že jste ho očesali o přihlašovací údaje pro mysql server. (z bezpečnostních důvodů)

Jedná se o tento odstavec ( který jsem vyjmul z oficiálních materiálů trinity) :

Jen pro vysvětlení:

ip adresa mysql serveru; port;jméno;heslo;jméno databáze

```
LoginDatabaseInfo      = "127.0.0.1;3306;trinity;trinity;auth"
```

```
WorldDatabaseInfo      = "127.0.0.1;3306;trinity;trinity;world"
```

```
CharacterDatabaseInfo  = "127.0.0.1;3306;trinity;trinity;characters"
```

Další klíčová věc, kterou zjistíme v databázi auth neboli realmd je realmd id (je to v tabulce realmlist)

RealmID = 1 je id Live serveru

RealmID = 2 je id Devs only

RealmID = 3 je id Chudáka.

Doporučuji v tomhle mít pořádek, jinak hrozí problémy s tím, aby to vůbec naběhlo.

Další specifická nastavení už stačí vzít z předchozích revizí po předchozích developerech, nebo si to jednou za čas celé projít, občas si zagooglit co která možnost způsobuje a co by

bylo dobré tak či onak nastavit a informovat předtím než takovou nějakou změnu uděláte určitě admina, případně i hlavní Gmko, aby o tom věděli 2 lidé, nebo případně další kolegy developery z dev teamu.

## Lekce 4

### ***Git, jak stáhnout vlastní zdrojáky a skompilovat si je podle návodu na Trinity core 2***

Pokud chcete být špičkový developer, ušetří vám hodně času, když se naučíte si prvně skompilovat nějakou zkušební revizi u sebe doma, než to zkusíte nahodit na ostro, nebo alespoň na test server. A taky doporučuji si to odladit doma, pokud máte nějaký výkonnější stroj, který rychle kompiluje, ať nemusíte zbytečně čekat na serveru, až se něco dokompiluje a ztrácet tak drahocenný čas. Na googlu je kopec tutoriálů, věřím že i na youtube najdete videotutorialy jak si trinity core2 rozběhat.

Github je výborná utilitka pro udržování nejaktuálnější verze zdrojového kódu a ušetření si 60% prací developera, protože si můžete vytvořit vlastní github úložiště, do kterého nahrajete zdrojové kódy, můžete tam vždycky něco dohrát vlastního za zdrojáky, online si to editovat a zároveň můžete synchronizovat další zdrojáky přímo z trinity. Přiznám se, že já jsem stará škola, takže za nás jestli tohle už bylo, tak to tenkrát ještě nikdo nepoužíval, nebo o tom nikdo nevěděl, ale teď už se to používá ve velkém a důrazně to doporučuji každému developerovi začít používat, šetří to čas, práci a nervy.

Zde je link na úložiště revizí trinitycore2 projektu :

<https://github.com/trinitycore2/trinitycore>

Github je výhodný v tom, že jedním příkazem lze všechno sesynchronizovat, jedním příkazem si můžete všechno stáhnout, ale jedním příkazem můžete taky všechno podělat a smazat...

Myslím, že jednodušší pro mě bude když sem spíš naházím odkazy a vložím návod Sholta, který vytvořil před dávnějšími časy pro 3server : (bohužel ho však taky nedodělal)

# 3server vytvorenie serveru, zaobchádzanie s gitom, updatovanie

Na úvod chcem podotknúť, že veľa vecí stále neviem, s gitom nerobím dlho, takže budem návod postupne dopĺňať.

## Windows

1. Potrebný software
2. Inštalácia
3. Kompilácia
4. Databáza
5. Konfigurácia

## Linux

1. Potrebný software
2. Inštalácia
3. Kompilácia
4. Konfigurácia

## Windows

### 1. Potrebný software

Linky:

3server-TCore: <http://codaset.com/msholto/3server-tcore/source>

Trinity-Core: <https://github.com/TrinityCore/TrinityCore>

Trinity-Db: <http://code.google.com/p/trinitydb/source/list>

Lazzalf-Core: <http://code.google.com/p/lazzalf-trinitycore/source/list>

Bloody-Core: <https://github.com/A-Metaphysical-Drama/BloodyCore/commits/>

Git 1.7.4. : <http://code.google.com/p/msysgit/downloads/detail?name=Git-1.7.4-preview20110204.exe&can=2&q=>

MySysGit: <http://code.google.com/p/msysgit/downloads/detail?name=msysGit-fullinstall-1.7.4-preview20110204.exe&can=2&q=>

Tortoise Git: <http://code.google.com/p/tortoisegit/downloads/detail?name=TortoiseGit-1.6.3.0-32bit.msi&can=2&q=>

Tortoise HG: <http://tortoisehg.bitbucket.org/>

Visual Studio 2008(vcsetup.exe) :

<http://www.microsoft.com/downloads/en/details.aspx?familyid=F3FBB04E-92C2-4701-B4BA-92E26E408569&displaylang=en#filelist>

MySQL: <http://dev.mysql.com/downloads/mysql/>

MySQL Connector: <http://dev.mysql.com/downloads/connector/c/>

Net Framework: <http://www.microsoft.com/downloads/en/details.aspx?FamilyId=333325FD-AE52-4E35-B531-508D977D32A6&displaylang=en>

SqlYog: <http://code.google.com/p/sqlyog/downloads/list>

Cmake (win32 installer) : <http://www.cmake.org/cmake/resources/software.html>

OpenSSL: [http://www.slproweb.com/download/Win32OpenSSL-1\\_0\\_0c.exe](http://www.slproweb.com/download/Win32OpenSSL-1_0_0c.exe)

Visual Studio C++ 2008 SP1 Redistributable Package:

<http://www.microsoft.com/downloads/en/details.aspx?familyid=A5C84275-3B97-4AB7-A40D-3802B2AF5FC2&displaylang=en>

Truice: <http://code.google.com/p/truice/downloads/list>

### 2. Inštalácia

Najprv si vytvorte zložku napr. „server“, kde budete mať všetko pohromade. Veci, ktoré s'ahujete a nemodifikujete inštalujte normálne do program files(defaultne) napr. Git, Tortoise, SqlYog, Cmake, OpenSSL, MySQL Connector , Visual Studio. To, kde modifikujete resp. ukladáte a meníte obsah si nainštalujte/ukladajte do zložky server.

Nainštalujte Git 1.7.4, MySysGit, Tortoise Git, Tortoise HG, Visual Studio, MySQL, MySQL Connector , Net

### 3. Kompilácia

#### GIT

Otvoríme zložku server(defaultne). Klikneme pravým tlačidlom myši na prázdnu plochu zložky a dáme „git clone“. Otvorí sa vám tabuľka programu Tortoise Git. Do kolonky „url“ vložíme **https** cestu k jadru, ktoré potrebujeme. Keďže je to pre 3server tak vložíme **clone**(clone adresa je na každej stránke git jadra napr. adresa <https://codaset.com/msholto/3server-tcore.git>. Vždy musí mať na konci príponu git a začínať sa https. Ak pracujete cez putty a ssh, tak túto adresu nevyužijete.) adresu „<https://codaset.com/msholto/3server-tcore.git>“. Kolonku „directory“ program vyplní automaticky takže tam nič nemeníme a dáme OK. Následne uvidíte ako sa vám sťahujú data príslušného jadra. Ak sa nejde pripojiť na stránku skontrolujte clone adresu, firewall alebo užívateľské práva. Po stiahnutí dát môžeme tortoise git clone okno zavrieť.

#### CMAKE

Následne spustíme program Cmake. Po spustení by vám malo vyhodiť okno pre vybratie kompilátora(ak nie tak ho spustíte v tools->configure). Keďže máme Visual Studio 2008 tak zvolíme to(Visual Studio 9 2008). **Nevyberajte možnosť Win64 ak máte Express verziu Visual Studia!** Necháme začiarknuté „Use default native compilers“. Ak napíše chybu „Error in configuration progress, project files may be invalid“ tak je to preto, lebo nemáme zadanú cestu k jadru. Takže vyplníme kolonku „Where is the source code:“ cestou k 3server-tcore jadru napr. „C:/server//3server-tcore“. Do kolonky „Where to build the binaries“ zadáme tú istú cestu, ale dopíšeme za „3server-tcore“ slovo „bin“ napr. „C:/server//3server-tcore bin“. Dole vľavo klikneme tlačidlo „Configure“. Ak nevypíše chybu tak sa ukáťu nastavenia kompilácie. Začiarknuté musíte mať: Scripts, Servers, Tools, Use\_corepch, use\_mysql\_sources, use\_scriptpch, with\_warnings. Ak niečo z toho nemáte začiarknuté, tak to začiarknite. Naopak ak máte začiarknuté niečo navyše, tak to odčiarknite. Klikneme znovu „Configure“ a po nakonfigurovaní klikneme na tlačidlo „Generate“.

#### VISUAL STUDIO

Následne otvoríme program Microsoft Visual Studio 2008 Express Edition. Dáme File->Open->Project/Solution a nájdeme súbor „TrinityCore.sln“ v zložke „3server-tcore bin“ a dáme open. Na hornej lište zmeníme „debug“ na „release“ a dáme build->build solution. Proces dost' zaťažuje PC. Doporučujem vypnúť čo najviac programov. Po dokončení(X success, 0 failed ...) nám vzniknú nové súbory v zložke „3server-tcore bin/bin/Release“ worldserver.exe, authserver.exe, ace.dll, authserver.conf.dist(príponu „.dist“ odstráňte buď cez total commander, alebo si v možnostiach zložky nastavte zobrazovanie prípon), worldserver.conf.dist(príponu „.dist“ odstráňte buď cez total commander, alebo si v možnostiach zložky nastavte zobrazovanie prípon). Knižnicu(.dll prípona) libmysql.dll získate zo zložky inštalovaného mysql connectoru.

### 4. Databáza

TrinityCore používa databázu Sql. Najprv potrebujeme takúto databázu získať. Toto je čistá databáza TrinityCore „<http://code.google.com/p/trinitydb/source/list>“. **POZOR! code.google.com ako aj bitbucket používa mercurial(HG) systém.** Ak chceme len plnú databázu, tak otvoríme kolonku „downloads“ a stiahneme databázu s najväčším číslom revízie(revízia je číslo, ktoré zahŕňa určitú zmenu. T.j. 1 revízia = 1 zmena) napr. 11808(11808 zmien). Rozbalíme vo winrar a nahráme cez SqlYog. 3server súčasne používa YTDB pre TRINITY „<https://bitbucket.org/dev1/ytdb-for-trinity/overview>“. Stiahnete ju pomocou tlačidla „Get Source“ na webe. Vyberete .zip, stiahnete, rozbalíte, nahráte.

#### NAHRÁVANIE

Po úspešnej konfigurácii MySql(next, next, next, ..., nastavenie portu, nastavenie hesla, start mysql(ak vyhodilo chybu, že nemôže naštartovať/ aplikovať, tak restartujte počítač a spuťte to znovu)) spustíme SqlYog. Pripojíme sa na MySql(localhost = 127.0.0.1 = Vaše PC). Pravým klikneme vľavo na pripojenie a vytvoríme 3 nové databázy s názvami: World, Characters, Auth. Nasledovne nahráme do characters(pravým klikneme na príslušnú databázu a vybereme možnosť „restore from sql dump“) súbor „characters\_database.sql“ v zložke „3server-tcore/sql/base“. To isté spravíme s databázou Auth, do ktorej ale nahráme „auth\_database.sql“ v tej istej zložke. World databáza je zložitejšia. Najprv nahráme Full\_DB, ktorú sme stiahli(vačšinou býva zabalená(.rar)). **Miešanie DB je na vlastné riziko!** Ak pri DB sú nejaké update packy(menší vydanie k db, ktoré sa ale nahráva do hlavnej), tak ich nahráme.

Nasledovne cez slyog vyberem zložku „3server-tcore/updates“ a nahráme všetky .sql súbory v tej zložke(old zložky nenahrávame) do príslušnej db.

## 5. Konfigurácia

Všetky konfigurácie sú nastaviteľné v súboroch worldserver.conf a authserver.conf. Keď ich otvoríme, tak vidíme, že niektoré riadky začínajú znakom # a iné nie. Ten znak znamená, že ten riadok sa nebude brať do úvahy, takže tam môžeme napísať hocičo(napr. aké nastavenie slúži na čo). Nás ale zaujíma základné nastavenie pripojenia jadra k MySQL. Nájdi riadky bez znaku #:

```
LoginDatabaseInfo = "localhost;3306;root;trinity;auth"  
WorldDatabaseInfo = " localhost;3306; root;trinity;world"  
CharacterDatabaseInfo = " localhost;3306; root;trinity;characters"
```

A prepíšte heslo(trinity) na heslo, ktoré ste zadali a pripájate sa cez neho do MySQL. To isté spravte so súborom authserver.conf, kde je riadok:

```
LoginDatabaseInfo = " localhost;3306;root;trinity;auth"
```

## Linux

### 1. Potrebný software

Linky:

3server-TCore <http://codaset.com/msholto/3server-tcore/source>

Trinity-Core <https://github.com/TrinityCore/TrinityCore>

Kompilačné súbory, cmake, git-core, mercurial:

```
sudo apt-get install build-essential autoconf libtool gcc g++ make cmake git-core mercurial patch wget links zip  
unzip unrar
```

OpenSSL, MySQL, LibMySQL:

```
sudo apt-get install openssl libssl-dev mysql-server mysql-client libmysqlclient15-dev libmysql++-dev libreadline5-  
dev zlib1g-dev libbz2-dev
```

ACE:

```
wget http://download.dre.vanderbilt.edu/previous\_versions/ACE-5.7.8.tar.gz
```

### 2. Inštalácia

Na úvod sa musíte oboznámiť s príkazmi, keďže linux beží hlavne na príkazovom riadku.

Príkazy a hodnoty sa nezadávajú v úvodzovkách, linux systém rozlišuje malé a veľké písmená.

sudo = príkaz ide cez roota(správca)

sudo su = prihlásenie za roota na stálo(vždy, ak nie ste prihlásený za roota, použite)

sudo chmod 777 „súbor“ = určený súbor získa povolenie pre zapisovanie

apt-get install „software“ = stiahne a nainštaluje software

cd „zložka“ = vojdete do zvolenej zložky

cd.. = vstúpíte o zložku vyššie

git = príkazy pre git program

git clone „https/ssh adresa“ = stiahne zvolené jadro

git fetch = stiahne najnovšiu revíziu a mergne(spojí) najnovšiu a súčasnú revíziu

git pull = updatne zložku s revíziou(vždy najprv fetch, potom pull)

cmake = príkazy pre cmake

mkdir „názov zložky“ = vytvorí zložku

příkaz -parameter = pomocou znaku „-“, sa zadávajú parametre pre jednotlivé príkazy

Najprv otvoríme terminal. Následne napíšeme príkaz „sudo su“. Vypýta heslo, zadajte a ste prihlásený za roota(správca). Ak sme v zložke /home/vášnick tak napíšeme 2krát cd.. aby sme sa dostali na začiatok súborového stromu. Vytvoríme zložku trinity/trinity a to nasledovne:

```
mkdir trinity
```

```
cd trinity
```

```
mkdir trinity
```

```
cd trinity
```

Sme v zložke /trinity/trinity. Teraz musíme

copyright 3server.cz Autor developer Sholto

## Pro další zájem odkaz na zdroje v češtině a angličtině s komentářem

[http://wiki.wowresource.eu/index.php/TrinityCore\\_na\\_Linuxu](http://wiki.wowresource.eu/index.php/TrinityCore_na_Linuxu)

[http://www.trinitycore.info/Main\\_Page](http://www.trinitycore.info/Main_Page) odkaz na hlavní stránky wiki trinity core

tutorial jak vše rozběhat přímo zde <http://www.trinitycore.info/How-to:Linux>

Je to sice anglicky, ale příkazy jsou stejné, v případě nouze, pokud nejste Zdatní v angličtině, tak doporučuji kdyžtak alespoň, [http://translate.google.com/translate?js=n&prev=\\_t&hl=cs&ie=UTF-8&layout=2&eotf=1&sl=en&tl=cs&u=http%3A%2F%2Fwww.trinitycore.info%2FHow-to%3ALinux&act=url](http://translate.google.com/translate?js=n&prev=_t&hl=cs&ie=UTF-8&layout=2&eotf=1&sl=en&tl=cs&u=http%3A%2F%2Fwww.trinitycore.info%2FHow-to%3ALinux&act=url)

Ale pokud vám začne google překládat i syntaxe příkazů, tak si raději na chvílku vpravo nahoře překlikněte na původní text.

Na internetu existuje také spousta .sh skriptů, které jen spustíte a všechno ostatní udělají za vás, ale je dobré tomu prvně rozumět, vědět co který příkaz dělá.

Když vám něco někde vypíše nějakou hlášku, automaticky ji okamžitě nahažte do googlu a čtěte vysvětlení. Dělat developera je fakt snadná záležitost, není to tak o znalostech jako o přístupu a uvažování, při řešení problému. Dále pokaždé když něco neběhá koukejte do .log souborů, v nich máte vždycky jasně napsáno, co s tím je. Chybové hlášky vždycky skopírujte a vložte do googlu a uvidíte, že stejný problém, jako třeba zrovna řešíte vy měl i někdo jiný. Na googlu máte vše a díky googlu můžete dokázat to, aby náš server rozkvétal a zlepšoval se.

## Lekce 5

### Další osvěžení znalostí linuxu

Obsahem kapitoly bude utilita mc, příkazy ls, cd, mv, df -h, du -sh, pgrep, less, more, vim, nano, links, chmod, pstree, last, who, cp, w -u, , winscp, putty-gen, man, tail, head.

Utility u klienta : winscp, putty-gen, putty,

Informující příkazy : ls, df -h, du -sh, pgrep, pstree, tail, head, less, more, last, uptime, who, w -u (tyto příkazy jenom něco vypisují a nic nemění)

Náhrazka total commanderu v příkazové řádce : mc

Příkazy, které něco dělají ( nebo jsou schopny změnit) : cd, mv, vim, nano, links, chmod, cp

Když nevíte co dělá daný příkaz, dejte si man příkaz a objeví se vám manuálové stránky pro příslušný příkaz a všechny podrobnosti a přepínače příkazu. Vyjedete z tím :q nebo jen zmáčknutím q.

VIM a NANO jsou textové editory. NANO Je docela intuitivní a VIM velice kvalitní informace ZDE :<http://www.nti.tul.cz/~satrapa/docs/vim/>

Všechny tyto příkazy lze nalézt na [www.linuxvbashi.cz](http://www.linuxvbashi.cz) a vyzkoušet si na Chudákovi .

Top a Htop – vyjedete z nich q.

Příkaz CHMOD je velice důležitý. Jedná se totiž o změnu přístupových práv souboru nebo adresáře. Na linuxu má každý soubor a adresář svoje přístupová práva vlastníka, skupiny a ostatních. Když nedáte přístupová práva pro skupinu ani pro ostatní, tak můžete do vašeho adresáře chodit jen vy a root (administrátor). Práva mají své hodnoty. Například CHMOD 777 soubor znamená že soubor mohou číst , zapisovat do něj a spouštět naprosto všichni.

Chmod XYZ soubor X jsou práva vlastníka, Y skupiny, Z ostatních.

V podstatě se to počítá, takže CHMOD 777 znamená všichni mohou zapisovat, číst, spouštět. 1 = spouštět , 2 = zápis, 4 = číst. Potom tedy  $1 + 2 + 4 = 7$  = všechna práva (spouštět, zapisovat, číst). Lze to i kombinovat. Takže například  $1 + 4 = 5$  tedy práva spustit a číst.

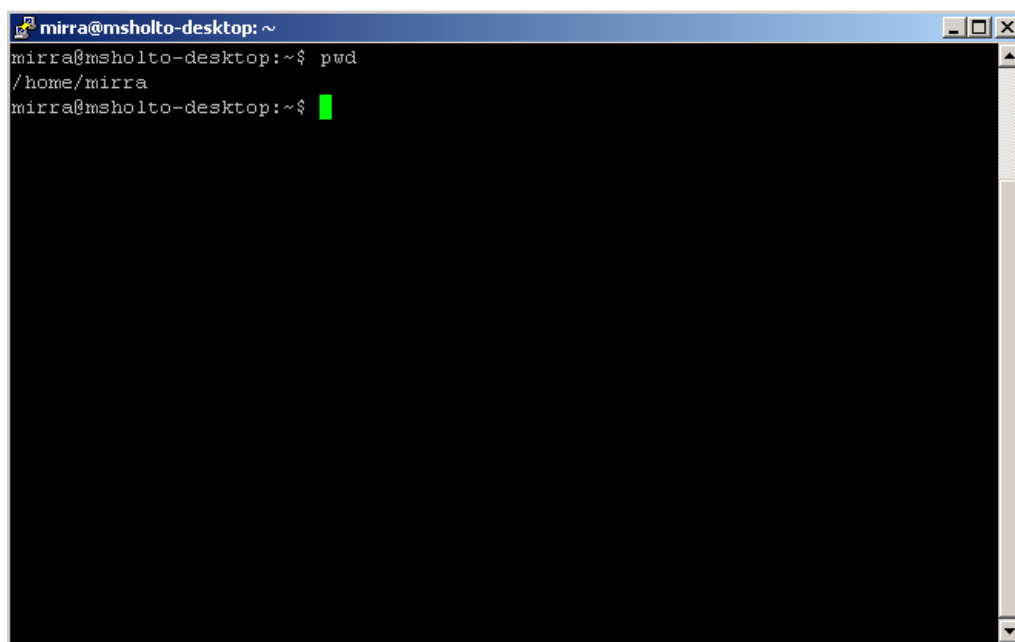
Příkaz chmod se tedy používá hlavně tehdy, když chcete zajistit, aby některý proces mohl spustit jiný uživatel, nebo aby třeba logy mohli číst všichni i ti, kteří třeba nemají oprávnění admina apod...

Více na linuxvbashi.cz a zde <http://cs.wikipedia.org/wiki/Chmod>

Začneme úplnými základy :

Předpokládejme že už zvládáme přihlášení na server a že jsme na něm přihlášení.

`pwd` - zobrazí cestu, ve které se právě nacházíme. Například :



```
mirra@msholto-desktop: ~  
mirra@msholto-desktop:~$ pwd  
/home/mirra  
mirra@msholto-desktop:~$
```

`cd ..` - posunete se o jeden adresář výš.

```
mirra@msholto-desktop: ~  
mirra@msholto-desktop:~$ pwd  
/home/mirra  
mirra@msholto-desktop:~$ cd ..  
mirra@msholto-desktop:/home$ pwd  
/home  
mirra@msholto-desktop:/home$ cd mirra/  
mirra@msholto-desktop:~$ cd  
.mc/ baf/ logy/  
mirra@msholto-desktop:~$ cd  
.mc/ baf/ logy/  
mirra@msholto-desktop:~$ cd  
.mc/ baf/ logy/  
mirra@msholto-desktop:~$ cd  
.mc/ baf/ logy/  
mirra@msholto-desktop:~$ cd  
.mc/ baf/ logy/  
mirra@msholto-desktop:~$ cd <tabulator>
```

*cd jméno adresáře* – posunete se o jeden adresář níž do konkrétního jména adresáře

O příkazu ls zde : <http://www.volny.cz/nuclearmike/linux/2/ls.html>

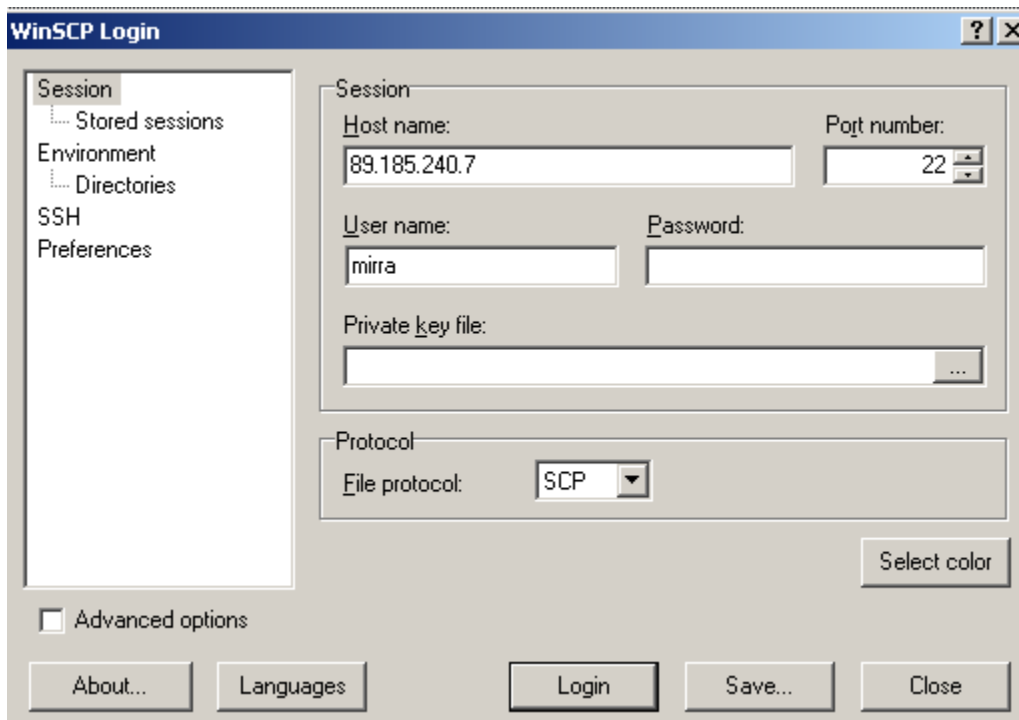
Tabulátor v linuxu slouží k doplňování názvů. Takže stačí napsat v příkladu třeba cd mi<tabulátor> a buď vám linux nabídne více možností, pokud je jich víc než 1, anebo vám to doplní název.

## **WINSCP**

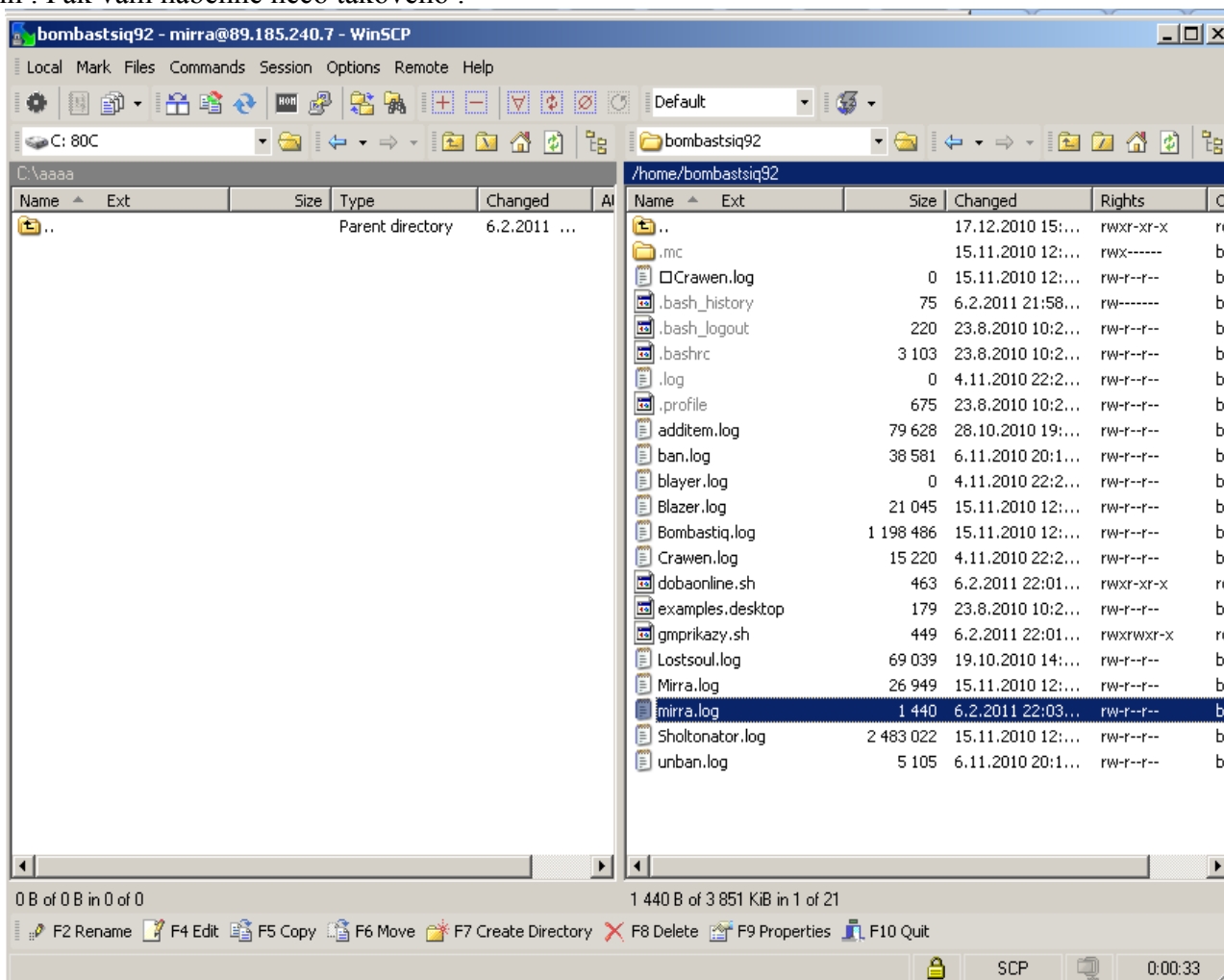
V podstatě to slouží jako FTP klient, ale běhá to přes SSH. Běží to na Windows a pod WINE na linuxu. Existuje jak instalovatelná verze, tak portable verze bez nutnosti instalace.

Když program spustíte, vypadá nějak takto :





Vyplníte potřebné údaje (username, password, případně pokud jdete přes RSA autentifikaci tak private key file) a dáte login . Pak vám naběhne něco takového :



Po připojení už se to chová jako nějaký total commander, či salamander nebo jiný souborový manažer. F5 překopírujete cokoliv do/ze svého adresáře ve vašem počítači a máte klid.

Pokud se budete muset připojovat na server (a to hlavně na 3server.net = hlavní stroj, tak budete potřebovat mít

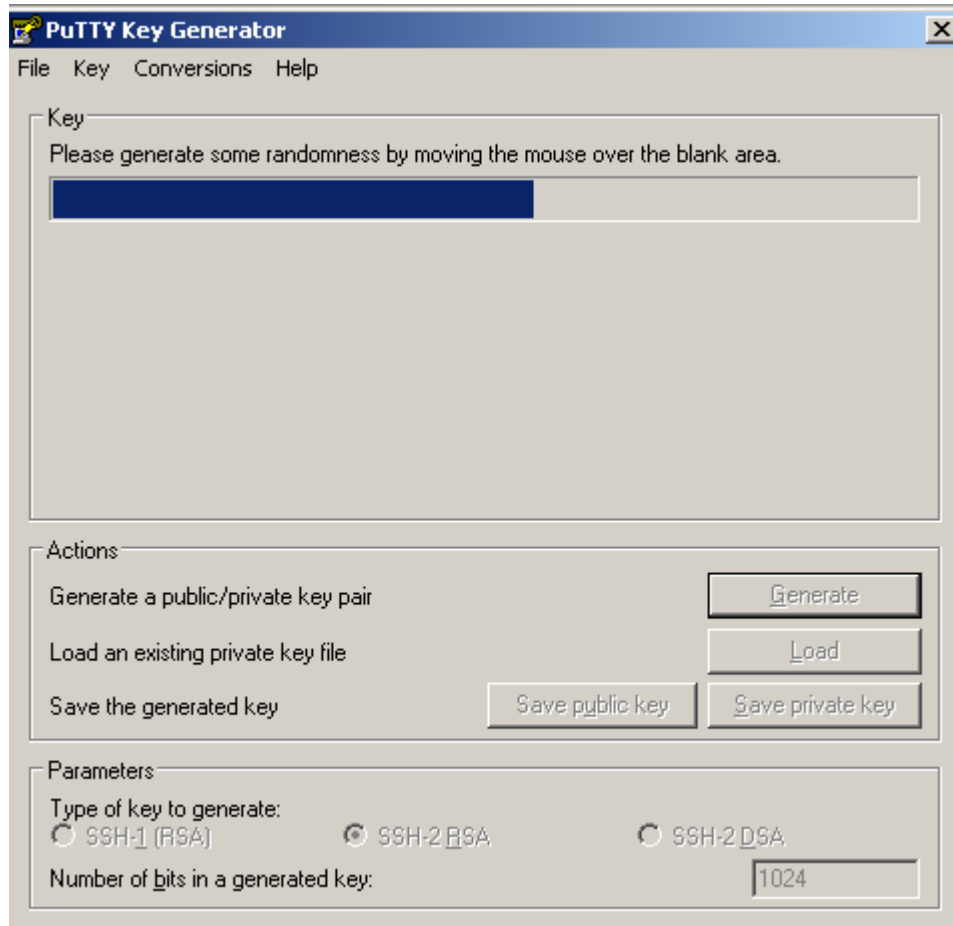
vygenerovaný RSA certifikát, bez kterého se na roota na hlavním stroji nepřihlásíte)

## ***Jak se připojit na server pomocí SSH certifikátu s RSA zabezpečením***

### **Jak vygenerovat vlastní SSH klíč:**

1) stáhni program puttygen.exe třeba na <http://tartarus.org/~simon/putty-snapshots/x86/puttygen.exe>

2) spusť a stiskni generate, pak chvíli hýbej s myší, dokud ti pole s „loadingem“ nedoběhne úplně do prava, jak je vidno na Obrázku č.1 ukázka generování SSH-RSA2 klíče.

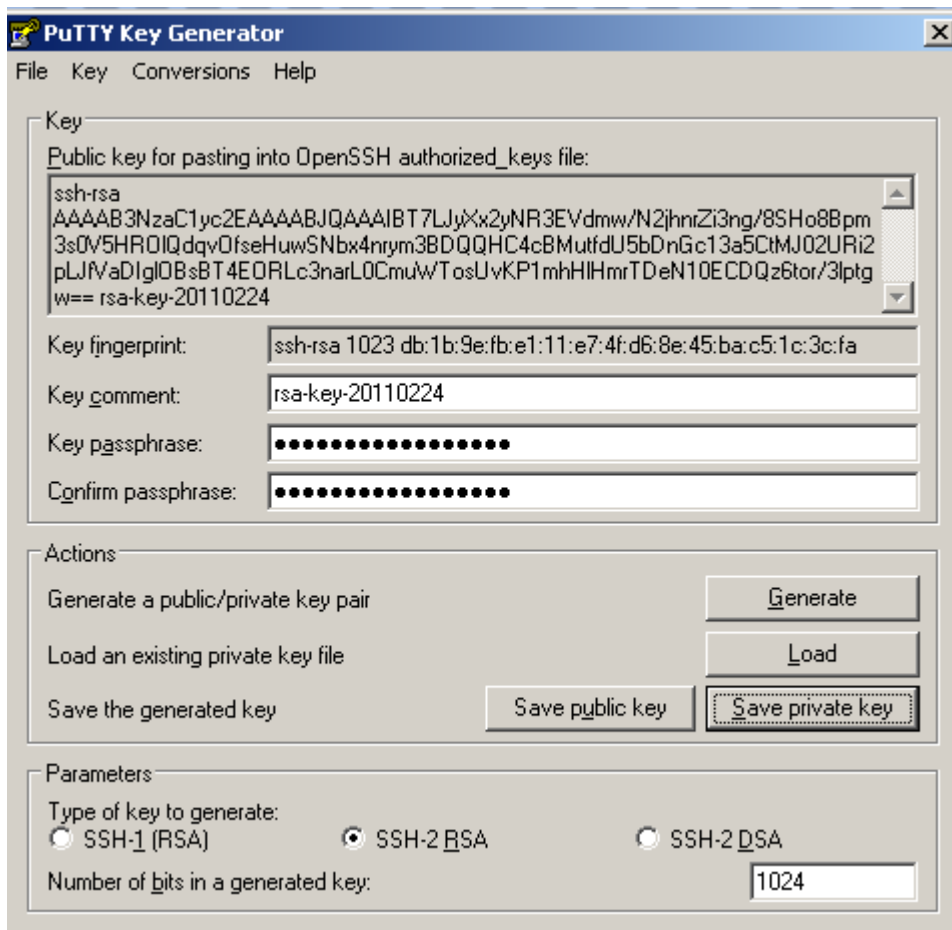


Obrázek č.1 ukázka generování SSH-RSA2 klíče.

3) Až ti to doběhlo do 100% tak do pole key passphrase zadej svoje heslo, které nikomu nikdy neřekneš (jen pokud by to byl mafián, který by na tebe mířil pistolí a ani to nikomu neřekni, dokud ti nezačne řezat důležité části lidského těla) a to stejné i do políčka Confirma passphrase.

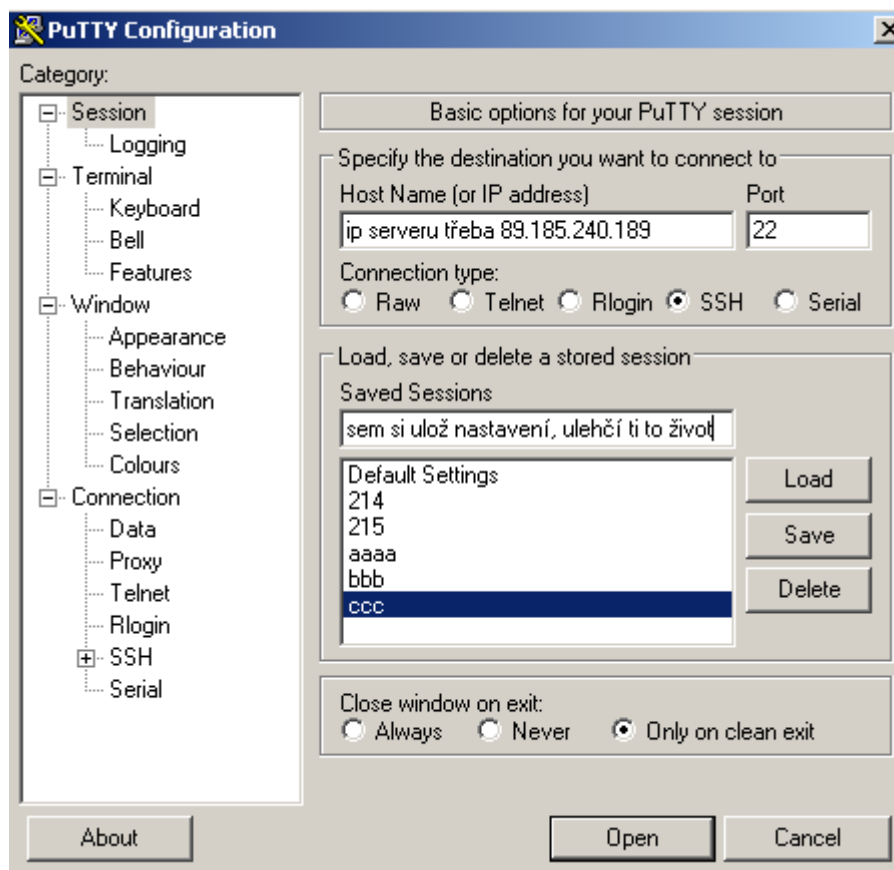
4) Až máš vyplněno, ověř si, že máš dole vpravo hodnotu 1024 (= 1024bitové kódování) zaškrtnutou hodnotu SSH-2 RSA.

Pak dáš Save public key a Save private key a máš hotovo podle obrázku Obrázku č.2 vyplnění potřebných údajů pro úspěšnou tvorbu SSH-RSA2 klíče.

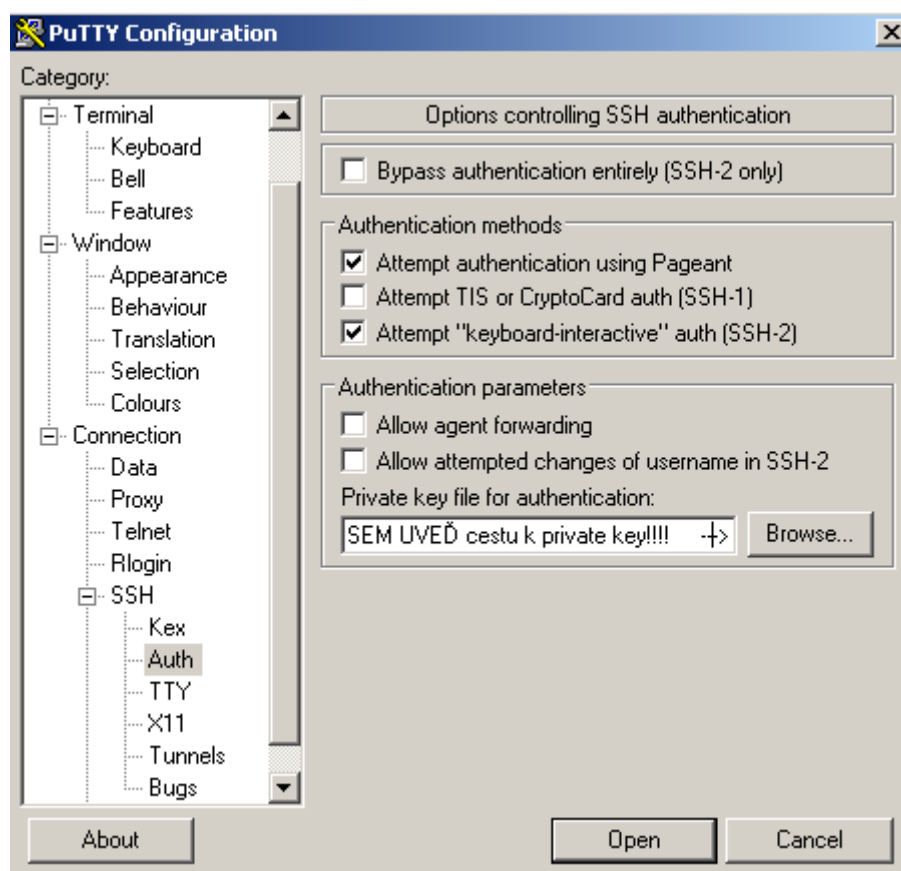


Obrázek č.2 vyplnění potřebných údajů pro úspěšnou tvorbu SSH-RSA2 klíče.

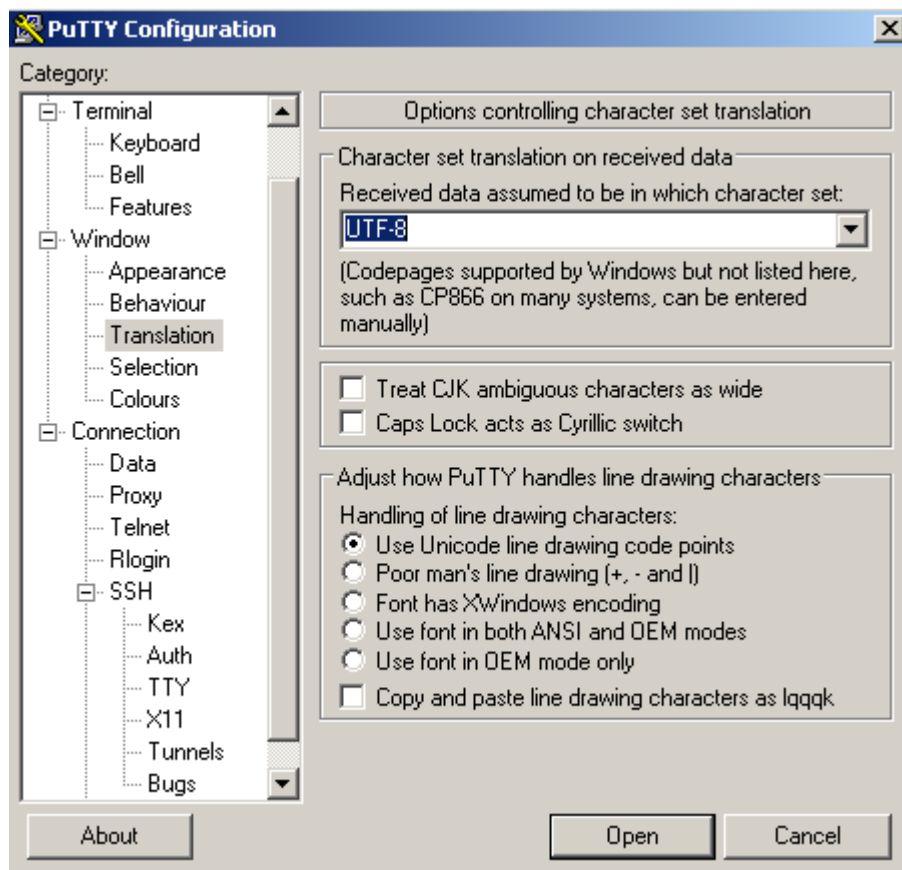
## Jak se připojit s SSH certifikátem



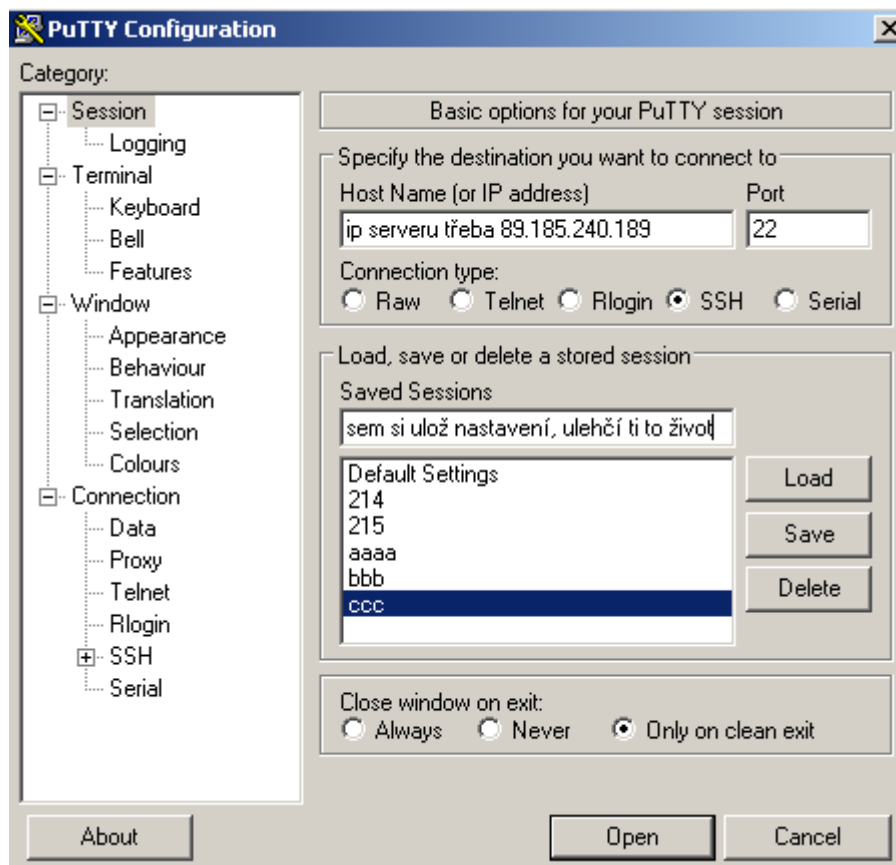
1) Vyplň v Host Name zadej ip serveru.



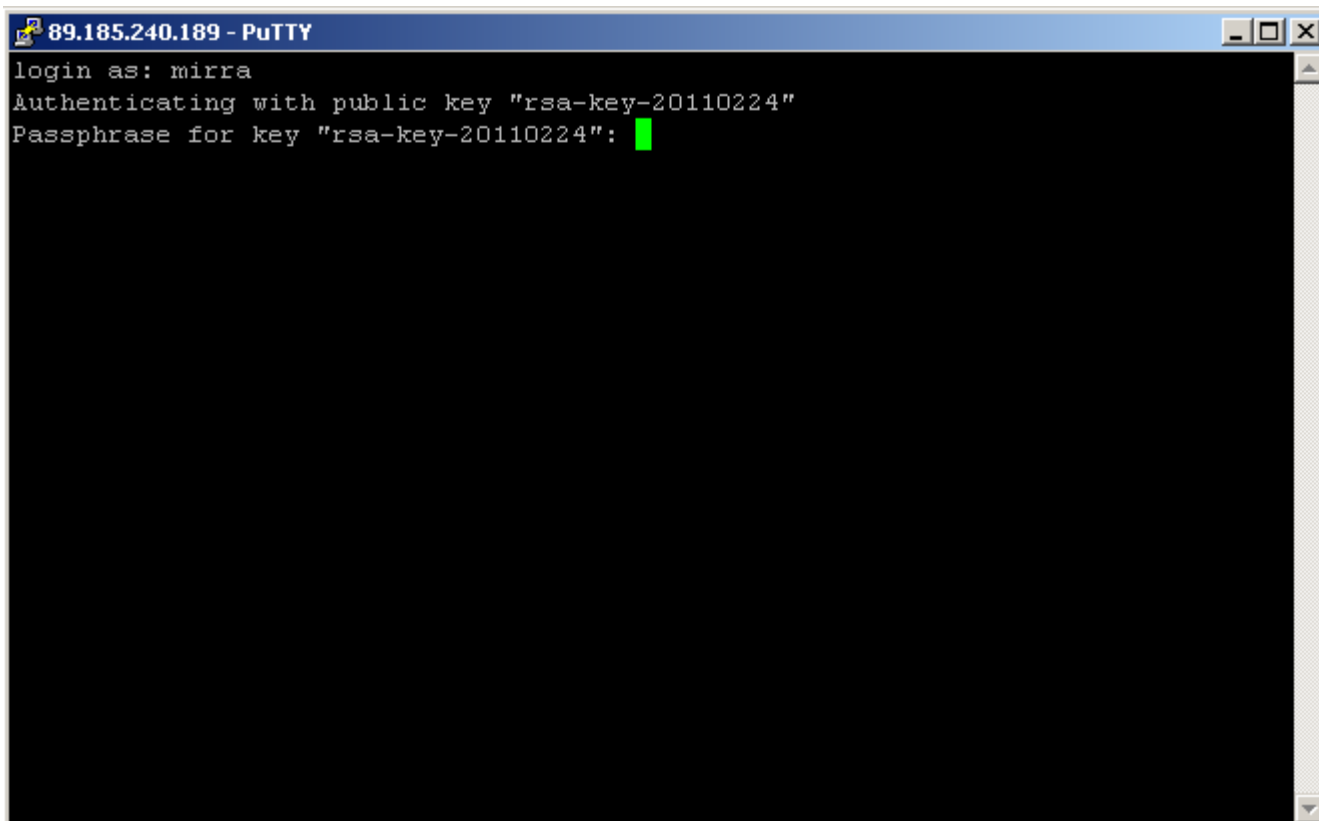
2) V levém menu si rozklikni SSH a klikni na Auth. Klikni na browse a vyber si z disku svůj ssh certifikát, který jsi před chvílí vygeneroval/a.



3) Klikni na Translation a ve vysouvacím menu Received data assumed to be in which character set: vyber UTF-8



4) Vrať se do Session (úplně nahoře v levém menu) a klikni na připojit se a jsi tam. Pokud ti to nenapiše Server Refused our key, tak je vše v pořádku. Zadáš heslo a jsi tam. Komunikace mezi tebou a serverem je nyní bezpečná.



89.185.240.189 - PuTTY

```
login as: mirra
Authenticating with public key "rsa-key-20110224"
Passphrase for key "rsa-key-20110224": █
```

Copyright 3server.cz Bez souhlasu vlastníka nelze kopírovat. Pro udělení souhlasu práv stačí odeslat email na [admin@3server.cz](mailto:admin@3server.cz) nebo na [3server@seznam.cz](mailto:3server@seznam.cz)