

04 - Moduly I.

Image: `y36aws-apache-04-mod0910`

- Cíle cvičení:**
- procvičit si konfiguraci virtualhostingu - opravit chyby
 - konfigurace logování - `ErrorLog`, `syslog`, `CustomLog`
 - `aws-common`, `aws-referer`, `aws-combinated`,
 - `Multi-Processing Modules`
 - modul `worker`
 - `mod_status`, `mod_info` – načtení, konfigurace

Změny a opravy v image

- tímto cvičením počínaje se kompletně využívá balíku `>=www-servers/apache-2.2.9`, to znamená:
 - spouštění, zastavování serveru a test konfigurace pomocí init skriptu `/etc/init.d/apache2`,
 - konfigurace init skriptu je uložena v souboru `/etc/conf.d/apache2`,
 - podstatná je proměnná `APACHE2_OPTS`, kde lze zadat parametry příkazového řádku (`-D`),
 - konfigurace serveru je v adresáři `/etc/apache2`,
 - hlavní konfigurační soubor je `/etc/apache2/httpd.conf`.
- soubor `/etc/hosts` je generován automaticky,
- do proměnné `$no_proxy` byly přidány všechny používané `*.k328` doménové názvy, není již třeba měnit nastavení Firefoxu,
- na učitelském počítači se používá driver grafiky `vesa`, rozlišení je kvůli projektoru 1024×768.

Virtualhosting

1. Ověřte, že fungují následující virtuální hostitelé a zobrazuje se požadovaný obsah.
 - *Nápověda* - pro `name-based` virtual hosting by měla být vyhrazena IP adresa `10.0.0.X-0.2`.
 - I. <http://www.mycorp.k328> a <http://www.aliascorp.k328> - zobrazí se stránka „Moje firma“
 - II. <http://www.othercorp1.k328> - zobrazí se stránka „Jiná firma 1“
 - III. <http://www.othercorp2.k328> - zobrazí se stránka „Jiná firma 2“
 - IV. <http://www.othercorp3.k328> - zobrazí se stránka „Jiná firma 1“ (ano, opravdu se má zobrazit „Jiná firma 1“)
 - V. <http://www.nocorp.k328> - zobrazí se stránka „Hlavní server“
2. Případné problémy diagnostikujte a opravte.

Nefunguje, byly nalezeny tyto chyby

- při startu apache zjistíme tuto chybu

- `NameVirtualHost 10.0.165.4:80` na `NameVirtualHost 10.0.165.2:80` v souboru `/etc/apache2/vhosts.d/20-name_othercorp1.conf`

- kontrola výpisem `netstat -tulpn` – na portu 80 nic neběží

- výpis z `/var/log/apache2/error_log` řekne, že nemůže načíst soubor `/etc/apache2/mime.types`

- výskyt najdeme příkazem `grep -R mime * -` hledá se v adresáři `apache2`
- zřejmě půjde o soubor `/etc/apache2/modules.d/...mime.conf`
- jde o `<IfDefine MIME>` = do souboru `/etc/conf.d/apache2` přidat `-D MIME` do položky `APACHE_OPTS`

- výpis z `/var/log/apache2/error_log` řekne, že nemůže načíst soubor `/tmp/mycorp/error_log`

- o v adresáři `/tmp/` vytvořit adresář `/mycorp/`
- přidat položku `ServerAlias www.othercorp3.k328` do souboru `/etc/apache2/vhosts.d/20-name_othercorp1.conf`
- v souboru `/etc/apache2/vhosts.d/...default...conf` opravit `Listen 8080` na `Listen *:80`

Nyní fungují všechny výpisy dle zadání.

Logování

ErrorLog

Konfigurace je v souboru `/etc/apache2/modules.d/00_default_settings.conf`

1. Pro celý server nastavte **ErrorLog** do souboru `/var/log/apache2/main_error_log`.

V konfiguráku `/etc/apache2/modules.d/00_default_settings.conf` přepsat `error_log` na `main_error_log` v pravidlu `ErrorLog`. Soubor se při testu vytvoří sám.

2. Pro virtuálního hostitele www.mycorp.k328 směrujte **ErrorLog** na **syslog**, facility **local3** (loguje do `/var/log/apache2-syslog/*error.log`)
 - o Do konfigurace logovacího daemona **syslog-ng** není třeba zasahovat, je připravena.

V konfiguráku `00-ip_mycorp.conf` přepsat `ErrorLog /tmp/...` na `ErrorLog syslog:local3`

3. Ověřte, že
 - o logování na **syslog** funguje
 - ověř se z URL – hledá se něco neexistujícího např. www.mycorp.k328/aaa + kontrola, že se vytvořil `/var/log/apache2-syslog/error_log` a obsahuje chybovou hlášku
 - o zprávy nejsou duplikovány v **ErrorLog** serveru.
 - kontrola standardního `error_log` souboru, že výše uvedený výpis není duplicitní

CustomLog

!!! Konfiguraci umístěte k hlavnímu serveru a k virtuálnímu hostiteli.

!!! Při logování do souboru (pokud se zprávy neobjevují a měly by) může být potřeba vynutit vyprázdnění bufferů příkazem `sync`.

1. Definujte následující formáty vlastních logů, jednotlivá pole oddělujte mezerou, pokud není řečeno jinak.

nastavení bude formou editu souboru `/etc/apache2/httpd.conf`

- o **aws-common** – klient = remote host = `%h`, vzdálený uživatel = remote user = `%u`, čas přijetí požadavku = `%t`, první řádka požadavku v uvozovkách `\"%r\"`, stav požadavku (po všech přesměrováních) = `%>s`, velikost odpovědi v bytech bez HTTP hlaviček = `%b`.

```
LogFormat "%h %u %t \"%r\" %>s %b" aws-common
```

- o **aws-referer** - Obsah hlavičky **Referer** odeslané serveru v požadavku, řetězec `: :`, požadovaná URL.

```
LogFormat "%{Referer}i::%U" aws-referer
```

- o **aws-combined** - spojení předchozích dvou formátů

```
LogFormat "%h %u %t \"%r\" \ %>s %b %{Referer}i::%U " aws-combined
```

2. Nastavte vlastní log tak, aby logoval do
 - o `/var/log/apache2/aws-access_log` ve formátu **aws-common**,
 - o `/var/log/apache2/aws-refer_log` ve formátu **aws-referer**.

konfigurace hlavního serveru = `/etc/apache2/vhosts.d/00-ip_default.conf`

do kontejneru `<VirtualHost _default_*>` přidat
`CustomLog /var/log/apache2/ aws-access_log aws-common`
`CustomLog /var/log/apache2/ aws-refer_log aws-referer`

3. Ve virtuálním hostiteli www.mycorp.k328 logujte rourou ve formátu **aws-combined**.
 - o Použijte nástroj `/usr/bin/logger`, který zprávy předává na **syslog**.
 - o Facility zvolte **local4** (výstup do souboru `/var/log/apache2-syslog/*_comb.log`).
 - o *Nápověda* - `/usr/bin/logger -p local4.0 -t apache2[combined]`

do `/etc/apache2/vhosts.d/05-ip_mycorp.conf` přidat
`CustomLog |"/usr/bin/logger -p local4.0 -t apache2[combined]" aws-combined`

4. Ověřte funkčnost logování.

Přes URL – napsat neexistující hledání + kontrola výpisu v logovacím souboru.

Zpracování logů

1. Napište shellový (případně jiný) skript, který do konzole vypíše počet přístupů k serveru z jednotlivých IP adres pro zadaný den.
 - o *Nápověda* - vytvořte si log, kde bude pouze čas ve vhodném formátu a IP adresy.

Zatím na to kašlem...

Multi-Processing Modules

Konfigurace je v souboru `/etc/apache2/modules.d/00_mpm.conf`

1. Seznamte se s konfigurací modulu **worker**
2. Nastavte jej tak, aby
= editace modulu **worker** v souboru `/etc/apache2/modules.d/00_mpm.conf`

- o najednou obsluhoval maximálně 100 požadavků,

```
MaxClients 100
```

- o ve stavu **idle** měl od 10 do 20 vláken

```
MinSpareThreads 10  
MaxSpareThreads 20  
ThreadsPerChild 20
```

3. Nastavení ověřte příkazem **ps** a výstupem z **mod_status** v dalším příkladu.

u příkazu **ps** jsme nepřišli na to, jaký zadat argument, aby byl výpis použitelný – viz **ps -help**

kontrola z `mod_status` `http://localhost/server-status`

mod_status, mod_info

1. Načtěte moduly.

Jsou už načtené v konfiguráku `httpd.conf` – jinak viz cviko 2.

2. Nakonfigurujte je tak, aby jejich výstup byl přístupný na `/status` a `/info` pro požadavky ze `10.0.0.0/8` a `127.0.0.0/8`.

v konfiguráku `/etc/apache2/modules.d/00_mod_info.conf` a `/etc/apache2/modules.d/00_mod_status.conf`

editovat

`<Location/server-info>` na `<Location/info>`

editovat a doplnit

```
Allow from 10.0.0.0/8
Allow from 127.0.0.0/8
```

3. Zorientujte se ve výpisu z `mod_info`, projděte si konfiguraci jednotlivých modulů.
4. Napište shellový (případně jiný) skript, který bude po dobu svého běhu do konzole vypisovat přenesený objem dat za určené období (např. každou minutu).
 - o Využijte výstupu `mod_status`.

Na to kašlem a na zajímavost taky... Zatím...

Zajímavost

Simulátor požadavků

Aby spojení bylo vidět ve výpisu `/server-status`, je potřeba, aby tvorba odpovědi serveru trvala delší dobu. To se dosáhne žádostí o `/cgi-bin/chunked.sh`, který trvá 5 s.

1. Vytvoření skriptu `/var/tmp/chnked.sh`

```
#!/bin/bash

echo "Content-Type: text/html"
echo
echo "<pre>"
for ((i=0; i<5; i++))
do
    date
    sleep 1
done
echo "</pre>"
```

2. nastavit přístupová práva pro skript (spouštění)
3. Nastavení serveru pro spouštění CGI skriptů `/cgi-bin/`

```
ScriptAlias "/cgi-bin/" "/var/tmp/"

<Directory /var/tmp>
    Order allow,deny
    Allow from all
</Directory>
```

4. Generátor požadavků

```
for ((i=1;i<=20;i++))
do
```

```
for ((j=1;j<=10*i;j++))
do
    wget -o /dev/null -O /dev/null http://127.0.0.1/cgi-bin/chunked.sh &
done
echo $i
wait
done
```

Generátor bude postupně posílat 10, 20, 30, ..., 200 požadavků najednou.

syslog-ng

Prostudujte si program **syslog-ng** a jeho konfigurační soubor. Zaměřte se především na část týkající se Apache httpd.

```
filter apache2-err      { facility(local3) and program("apache2"); };
filter apache2-comb    { facility(local4) and program("apache2"); };
destination apache2-err
    { file("/var/log/apache2-syslog/$YEAR/$MONTH/ $HOST_$DAY.error.log"); };
destination apache2-comb
    { file("/var/log/apache2-syslog/$YEAR/$MONTH/$HOST_$DAY.comb.log"); };
log { source(src); filter(apache2-err); destination(apache2-err);
    flags(final); };
log { source(src); filter(apache2-comb); destination(apache2-comb);
    flags(final); };
```