Chytáky aneb co v /etc/apache2/vhosts.d/* nebylo

Kukaččí httpd

Běžel tam cizí apache, kterej blokoval potřebný porty, toho potřeba killnout Stačilo příkazem.

```
/etc/init.d/apache22 stop
```

Běžel tam ještě jeden sirotek httpd, který poslouchal na adrese 172.16.X.9:80. Na ten se přišlo příkazem:

```
netstat -tanp      (neboli: --tcp --all --numeric --program)
```

A zabil se (za PID dosaď process id httpd daemona z předchozího příkladu).

```
kill PID
```

Dále tam ještě běžel apache21 tedy dokumentace, ta měla ještě nesmyslně nakonfigurované poslouchání na 172.16.X.[7-8]:80, ale tuším, že script, kterým kontrolovali naší práci zastavoval apache s dokumentací (volal následující příkaz). Já ho pro jistotu před odevzdáním vypnul:

```
/etc/init.d/apache21 stop
```

Chyby v konfigurácích

- /etc/apache2/httpd.conf - zakomentovaný load mod_alias
- /etc/apache2/modules.d/00_default.conf
  - ErrorLog /var/lock/apache2/error_log - lock misto log
  - .htaces misto .htaccess
  - misto index.html je default.html
  - jestli je tam nejaky defaultni nastaveni <Directory tak asi nejlip smazat.
- /etc/conf.d/apache2 - potřeba přidat definice do parametru pri startu "-D SUEXEC" (az nekdo zacne psat reseni otazek, tak tohle pridejte do reseni konkretnich bodu.. neni to chytak)

Troubleshooting

POKUD SI ZABIJETE DOKUMENTACI!

Například tím, že dáte `killall -9 httpd`, zkuste ji nastartovat příkazem:

```
/etc/init.d/apache21 restart
```

restart můžete obecně používat místo *start* jediný rozdíl je, že dostanete varování, když služba neběžela. Pokud vám init script tvrdí, že už běží a nechce se nastartovat dejte:

```
/etc/init.d/apache21 zap
```

to ho naučí, ale tuším, že by měl stačit restart.

Zadání

- rozběhání serveru
  - ifconfigem zjistit jaké je naše "X" z IP adresy =)
  - zabít konkurenci (viz. chytáky)
  - Listen na adresách 172.16.X.[1-4]
- name-based
  - dva hosti: othercorp1 a 2
  - jeden mel suexec
- ip based tusim mycorp (pise se primo jako parametr do <VirtualHost www.mycorp.k328>)
  - reverzni proxy
    - presmerovat /webmail nekam na othercorp1.k328/exter/webmail
    - presmerovat /neconeco na othercorp2.k328/neconeco
    - nevracet chybove stranky (4XX 5XX), ktere prijdou od reversovanych stranek, ale vracet vlastni (je to jedna direktiva).
- vlastni 404 chybova stranka
  - prenastavit 404 na nejakou vlastni stranku 404.html
  - pomoci type-mapy ji rozhodit na cs a en
  - pomoci SSI do ni vlozit datum, volanou adresu ktera vratila 404 a jmeno apache.

Y36AWS Praktická zkouška 21.1.2009

Z STM Wiki

Zkouška začla pár minut po 11, protože ještě "dodělávali" ty před náma. Všichni měli stejný zadání, některý kompy byly nefunkční. Dozor dělal Kadlec a Bařinka. Největším nepřítelem se však stal skript, který nakonec celou praktickou část zkontroloval. Znamená to, že body dostanete jen za kompletní řešení úkolu. Skript kontroluje výstupy, takže pokud máte třeba všechno nastaveno a chybí vám jen maličkost k tomu, aby to fungovalo, skript vám za úkol nedá ani bod. Existují již předudělané index.html stránky, které obsahují v komentáři nějaký hash. Zabijácký skript zřejmě kontroluje jestli se stránka zobrazila a porovná hash stránky. Na praktickou část máte tedy 3 hodiny - když si víte rady, dá se to zmáknout mnohem rychlejš. Skript vám pak během pár vteřin vyřkne ortel. Většina úkolů už je "přednastavena", takže žádný úkol nezačínáte psát od nuly. Většinou stačí někde něco přepsat a doplnit. Následně můžete jít psát teorii do vedlejší místnosti (průchozí, na chodbu se nutně nedostanete).

V prakticke casti byly dva bezici apache, "apache21" a "apache22". Na "apache22" bezi dokumentace, nicemu neprekazi, ale "apache21" je tam zbytecny.

Praktická část obsahovala 14 úkolů cca:

nastavení listenerů (8 bodů)

Pres "ifconfig" se clovek podiva, jake presne ip adresy ma. Pak Listen 1.2.3.4:80 pro kazdou pozadovanou adresu. Bylo nutne zrusit apache21, ponevadz blokoval jeden z interfaceu.

zařídit, aby se v odpovědi v hlavičce "Server" zobrazovalo pouze "Apache" (2 body)

Je potřeba nastavit direktivu:

```
ServerTokens Prod
```

nastavit default ip-based

```
<VirtualHost _default_>
```

nastavit nějaké alias adresáře

```
Alias /mycorp /var/www/vhosts/mycorp
```

userdir tak, že to bude public_html, ale když existuje adresář www, bude se jako userdir zobrazovat obsah www

To jsem resil tak, ze jsem udelal symlink public_html->www v kazdem z jednotlivych homediru.

Pak "UserDir public_html". Pozor, jsou umyslne rozbita prava k adresarum.

Mělo by to jít i takhle:

```
Userdir www public_html
```

aspoň podle dokumentace http://httpd.apache.org/docs/2.2/mod/mod_userdir.html

nastavení dvou name-based

```
NameVirtualHost 1.2.3.4
<VirtualHost default.vhost.com> ...
<VirtualHost dalsi.nedefault.vhost.com>
```

nastavení multiviews s languagepriority

```
Options +MultiViews
LanguagePriority cs en fr
```

filter podobný tomu ve cvičení, pro soubory .txt, .csv a .tex - převod malých a velkých písmen

```
ExtFilterDefine lowertoupper "/usr/bin/tr [a-z] [A-Z]" #Pozor, plna cesta k "tr", nehleda se v $PATH
AddOutputFilter lowertoupper txt csv tex
```

pro soubory .man nastavit typ text/plain

```
AddType text/plain .man
```

rozběhat skript ctverec.cgi, který vypočítá 2. mocninu z proměnné zadané v URL

Treba v PHP:

```
#!/usr/bin/php
<?php
echo "Content-type:text/html";
echo "\n\n";
$parts = explode("=", $_SERVER["QUERY_STRING"]);
$ctverec = pow($parts[1],2);
echo $ctverec;
?>
```

Pozn. v zadání bylo, že se má vzít číslo z proměnné "ctverec" - to je třeba ještě ošetřit, jinak předcházející skript naparsuje všechny proměnné.

Navrhuji následující:

```
#!/usr/bin/php
<?php
echo "Content-type:text/plain";
echo "\n\n";
$ctverec=ereg_replace(".*ctverec=([^&]*)(&.*)?","\\1",$_SERVER["QUERY_STRING"]);
echo pow($ctverec,2);
?>
```

Nebo v bashi:

```
#!/bin/bash
echo "Content-type:text/plain"
echo ""
a=`echo $QUERY_STRING|sed 's/.*ctverec=//'|sed 's/&.*//'`
expr $a '*' $a
```

Rewrite pro skript ctverec.cgi, /set/ctverec/<hodnota>.

```
RewriteRule ^/set/ctverec/([0-9]+)$ /cesta/k/tomu/kde/mate/ctverec.cgi?ctverec=$1
```

rozběhat suexec pro skript env.cgi

Presunout suexec v /usr/sbin a nastavit spravne prava (04750), uzivatele root a skupinu apache

Citováno z „http://stm-wiki.cz/index.php/Y36AWS_Praktick%C3%A1_zkou%C5%A1ka_21.1.2009“

Bylo to docela peklo, tak 2x těžší než minulá zkouška. Spousta věcí, co jsem si říkal že tam "nemůžou dát". Při přípavě je potřeba si udělat všechna cvičení i včetně proxy, loadbalancingu atd.

Co bylo rozbité:

- názvy ErrorLogu atd., to se dalo čekat
- hodně define modulů (i nesmyslně - například IfDefine auth_digest> LoadModule module_basic_auth modules/mod....)
- pro některé moduly úplně chyběla direktiva LoadModule
- rozbitá práva - nejenom u souboru ale i v nadřazených adresářích!

Co se mělo vyrobit:

Nastavení listen

Běžely tam dva apache, jeden s dokumentací a druhý blokoval potřebné IP adresy. Tak jsem jeden sestřelil a zrovna to byl ten s dokumentací :) Na spuštění pak stačilo spustit init skript /etc/init.d/apache21 (bez parametrů). Na zjištění který apache sestřelit je nejlepší použít příkaz netstat:

```
netstat -lpn
```

(l listen, p program, n numeric)

a potom zabit dany proces (dokumentace bezi na 8080 tak ten druhy, PID zjistime z nestats)

```
kill PID
```

Konfigurace mixed-hosts

Byl tam trochu chyták že jeden virtual host mohl být dostupný pouze z IP adresy 10.0.X.9 - to se z firefoxu neotestuje, je na to potřeba wget. Konkrétně:

```
wget --bind-address=
```

konfigurace load-balancingu pro jednoho vhosta - vhost měl fungovat jako reverzní proxy s loadbalancingem v poměru 1:5 na další dva virtual hosty, které běžely na serveru s dokumentací

```
<VirtualHost 10.0.130.4>
   ServerName www.proxy.k328
   ProxyRequests Off

   <Proxy balancer://mycluster>
       BalancerMember http://www.lb1.k328 loadfactor=1
       BalancerMember http://www.lb2.k328 loadfactor=5
       ProxySet lbmethod=bytraffic
   </Proxy>
   ProxyPass / balancer://mycluster
</VirtualHost>
```

CGI script, který vypočítá poměr komprese při použití mod_deflate, parametrem skriptu je url souboru, které se stáhne jednou s kompresí podruhé bez a poté se vypočítá se poměr komprese

Komprese se zajisti nastavenním SetOutputFilter DEFLATE pro všechny soubory v požadovaném adresáři

CGI skript:

```
#!/bin/bash
echo "Content-type:text/html";
echo
#zpracovani URL
url=`echo $QUERY_STRING | cut -d = -f 2`;
#stahovani
wget $url -O /tmp/file -q
withoutcompr=`du -b /tmp/file | cut -f 1`
wget --header='Accept-Encoding: gzip,deflate' $url -O /tmp/file -q
withcompr=`du -b /tmp/file | cut -f 1`
#vypocet
ratio=`expr $withoutcompr / $withcompr`
echo $ratio
```

Rewrite rule pro předchozí CGI skript

Chceme aby se napr. tato url:

```
http://localhost/cgi-bin/ratio/http://localhost/logs/error_log.log
```

prekladala na:

```
http://localhost/cgi-bin/compression_ratio.cgi?file=http://localhost/logs/error_log.log
```

Reseni:

```
RewriteEngine On
RewriteRule /ratio/(.+) /cgi-bin/compression_ratio.cgi?file=$1
```

Custom ErrorDocument 404 - podle jazyka klienta se zavolá

buď soubor 404.cs.html nebo 404.en.html, má to být realizováno pomocí type-map

Pro adresar, kde mame error dokumenty je potreba nasledujici konfigurace:

```
ErrorDocument 404 /error/404.var
<Directory "/var/www/localhost/error">
   Options +IncludesNoExec #pro SSI
   AddOutputFilter INCLUDES html #pro SSI
   AddHandler type-map var
   Order allow,deny
   Allow from all
   LanguagePriority cs, en
</Directory>
```

A type map soubor 404.var vypada takhle:

```
URI: 404.html
URI: 404.cs.html
Content-Language: cs-cz
Content-Type: text/html
URI: 404.en.html
Content-Language: en-us
Content-Type: text/html
```

V 404 dokumentu se má použít SSI pro výpis datumu

Obsah souboru 404.cs.html:

```
<html>
   <head>
      <title>404 Nenalezeno</title>
   </head>
   <body>
      <p>Tady nic neni</p>
      <p>Datum: <!--#echo var="DATE_LOCAL" --></p>
   </body>
</html>
```

V jednom virtual hostu nastavit pomocí .htaccess přístup do adresáře private přes HTTP Basic Auth se jménem Y36AWS, heslo Y36AWS

.htaccess soubor:

```
AuthType Basic
AuthName "Super tajny adresar"
AuthUserFile /var/www/localhost/passwords
Require valid-user
```

Soubor passwords se vytvori prikazem:

```
htpasswd -c /var/www/localhost/passwords y36aws
```

zablokování přístupu k souborům .htaccess

```
<FilesMatch "^\.ht">
      Order allow,deny
      Deny from all
</FilesMatch>
```