Apache savjeti i trikovi

Ova online knjiga je skup pojedinačnih članaka objavljenih kao pomoć sistemcima u konfiguraciji i korištenju apache web poslužitelja.

- Logirajte [1] se za dodavanje komentara

Apache2: Poruka "mixing * ports and non-* ports is not supported"

Svi iz iskustva znamo da ono što nije pokvareno ne treba popravljati. Sličan način razmišljanja su usvojili mnogi sistem inženjeri kod brige za svoje poslužitelje (što zapravo nije ispravan način razmišljanja, jer poslužitelji zahtijevaju konstantnu pažnju i kad se "ne bune", ali o tome drugom prilikom). Uzmimo za primjer web servis, koji se gotovo 100% izvršava pod najpopularnijem web poslužitelju - Apacheju. Iako inačica 2.0, pa i 2.2 postoje već jako dugo, mnogi su ostavljali Apache 1.3 u produkciji jer, eto,"sve radi". No, Apache 1.3 treba zaboraviti i krenuti dalje. Pri tome treba malo "stisnuti zube" i savladati manje poteškoće na putu.

Konfiguracija se Apache web poslužitelja prelaskom na Apache 2 ponešto izmjenila, te se ponekad mogu javiti manji problemi koji se u starijim inačicama nisu javljali. Ovo može biti jer su razviljatelji Apacheja odlučili onemogućiti neke stare *workaroundove*, ili jednostavno očistiti sintaksu od zaostataka starih vremena.

Ponekad (obično odmah nakon nadogradnje na Apache 2) u logovima možete naći ovakvu ili sličnu poruku:

```
[error] VirtualHost 193.198.X.Y:0 -- mixing * ports and non-* ports
with a NameVirtualHost address is not supported, proceeding with undefined results
```

Ova poruka je rezultat nepravilne konfiguracije virtualnih hostova baziranih na imenu (name-based vhosts). Primjerice:

```
NameVirtualHost 193.198.XXX.YYY:80

<VirtualHost www.domena.hr>
  ServerName www.domena.hr
  DocumentRoot /home/httpd/htdocs
  ErrorLog /var/log/apache/error.log
  TransferLog /var/log/apache/access.log
</VirtualHost>
```

Vrijednost NameVirtualHost i Virtualhost moraju biti identične, uključujući i broj porta. Apache 2 će uredno prijaviti problem, iako sam rad poslužitelja ne mora biti narušen, i svi virtualni hostovi mogu raditi bez ikakvih problema, no to ovisi o konkretnoj situaciji.

Ne ulazeći u sve moguće načine konfiguriranja Virtualnih hostova [2], navest ćemo primjer koji će odgovarati većini tipičnih konfiguracija na CARNetovim poslužiteljima:

```
NameVirtualHost 193.198.XXX.YYY:80

<VirtualHost 193.198.XXX.YYY:80>
  ServerName www.domena.hr
  DocumentRoot /home/httpd/htdocs
  ErrorLog /var/log/apache/error.log
  TransferLog /var/log/apache/access.log
</VirtualHost>

<VirtualHost 193.198.XXX.YYY:80>
  ServerName studenti.domena.hr
  DocumentRoot /home/httpd/htdocs
  ErrorLog /var/log/apache/error.log
  TransferLog /var/log/apache/access.log
</VirtualHost>

...itd
```

Dakle, preporučuje se rabiti IP adresa (jer ne zahtijeva DNS lookup) i eksplicitno navesti port na kojem će virtualni host slušati. Naravno, možete imati i druge vrijednosti za port, ali uvijek popraćen odgovarajućim NameVirtualHost direktivom (može ih biti više), npr. NameVirtualHost 193.198.XXX.YYY:8080.

Drugi slučaj koji se često javlja nije problem u konfiguraciji, ali se na prvi pogled tako može učiniti. Ako Vam se javi poruka:

```
Invalid command 'php_flag', perhaps mis-spelled or defined by a module
not included in the server configuration
invoke-rc.d: initscript apache2, action "start" failed.
```

Ovdje se radi o nedostatku paketa php4-cn, odnosno php5-cn. Nakon instalacije navedenog paketa, problem će nestati. Valja napomenuti da se greška može pojaviti prilikom nadogradnje sa Apacheja 1.3, a da problema zapravo nema. To se događa jer se kod nadogradnje događa nekoliko restarta i postoji mogućnost da u tom trenutku PHP još nije aktivan. Jednostavnim pregledom logova, odnosno provjerom vaših PHP stranica možete vidjeti da li sve radi, te tu poruku o greški zanemariti.

- Logirajte [1] se za dodavanje komentara

uto, 2008-09-30 22:03 - Željko Borš**Kuharice:** Za sistemce [3]
Kategorije: Servisi [4]
Vote: 0

No votes yet

Apache2: autentikacija korisnika rabeći samo LDAP

U člancima http://sistemac.carnet.hr/node/42 [5] (Autentikacija osnovnih servisa putem LDAP-a i RADIUS-a) i http://sistemac.carnet.hr/node/41 [6] (Autorizacija Apache web poslužitelja putem LDAP-a i RADIUS-a) smo objasnili kako autenticirati mrežne servise, s naglaskom na RADIUS. No, njegova uporaba nije nužna jer se sve može napraviti direktno preko LDAP-a.

Prvo, trebate osigurati da su učitani odgovarajući moduli, **mod_auth_basic** i **mod_authnz_ldap**. Možda ste primjetili da se modul ne zove mod_auth_ldap, nego mod_authnz_ldap. Ova se promjena dogodila od inačice 2.1, jer su neki moduli podijeljeni na autentikacijske ("n") i autorizacijske ("z"). Kako LDAP modul nosi obje oznake, znači da obavlja obje funkcije, što je svojevrstan povratak na staro. No, nećemo razbijati glavu s tim, nego idemo dalje.

Obično su svi potrebni moduli već učitani, ali ukoliko nisu, poslužite se naredbom **a2enmod**:

```
# a2enmod authnz_ldap
Module authnz_ldap installed; run /etc/init.d/apache2 force-reload to enable.
```

Nakon što ste poslušali savjet i restartali Apache na način koji se preporuča u poruci, možete dalje pristupiti konfiguraciji. Ona je zapravo vrlo jednostavna, sve što trebate učiniti je kreirati datoteku **.htaccess** koju ćete staviti u **direktorij koji želite zaštititi**. Ona ima ovakav sadržaj:

```
AuthType basic
AuthName "Molimo unesite LDAP zaporku"
AuthBasicProvider ldap
AuthLDAPURL ldap://ldap.domena.hr/dc=domena,dc=hr?hrEduPersonUniqueID
AuthLDAPRemoteUserIsDN off
require ldap-filter &(hrEduPersonUniqueID=*)
```

Što govore ove direktive? Situacija je prilično jasna, ove direktive određuju da će se rabiti osnovna ("basic") autentikacija (dakle, preko osnovnog HTTP Basic mehanizma), te određuju LDAP poslužitelj koji će se rabiti. Najzanimljiviji je parametar **ldap-filter**. On određuje koji će se atribut iz LDAP-a provjeravati, a u ovom slučaju je **hrEduPersonUniqueID**
upravo ono što nam treba, a to je AAI@EduHr korisnička oznaka.

Naravno, mogli ste staviti i samo "uid" umjesto "hrEduPersonUniqueID", što je skraćena inačica (umjesto "korisnik@domena.hr" pisat ćete samo "korisnik" u formu za prijavu).

Ovo može zbuniti korisnike, pa je možda bolje da ostavite "hrEduPersonUniqueID". Ne zaboravite isti atribut postaviti i u direktivi AuthLDAPURL. U ovom trenutku nećemo ulaziti u sve mogućnosti ovog modula, a za više informacija provjerite dokumentaciju na adresi:

```
http://httpd.apache.org/docs/2.2/mod/mod_authnz_ldap.html
```

- Logirajte [1] se za dodavanje komentara

čet, 2009-06-18 08:28 - Željko Borš**Kuharice:** Linux [7]
Kategorije: Servisi [4]
Vote: 5

Vaša ocjena: Nema Average: 5 (1 vote)

Apache2: kako ga instalirati?

 Nakon duljeg perioda izrade i testiranja paketa Apache 2, na CARNetovom repozitoriju paketa se pojavio paket apache2-cn za *Sarge* distribuciju. Preporučujemo da ovaj paket instalirate prije prelaska na Etch, jer stari Apache (inačica 1.3) neće više biti podržan. Iako je puno truda i vremena uloženo u izradu i testiranje ovog paketa (sto je rezultiralo s oko 1300 linija koda za prebacivanje konfiguracije s Apache 1 na Apache 2) i problema ne bi trebalo biti, molimo da svaki uočeni problem prijavite na e-mail adresu paketi@carnet.hr.

Nakon prelaska na Apache 2, stekli ste uvjete za upgrade na CARNet Etch distribuciju. Procedura prelaska će biti objavljena nakon što procjenimo da je dovoljan broj institucija prešao na Apache 2.

Postupak nadogradnje je sljedeći:

```
# apt-get update
# apt-get dist-upgrade
# apt-get install apache2-cn
```

Napomena: kod nekih poslužitelja je dovoljno i samo "apt-get upgrade", ali je oblik "apt-get dist-upgrade" preporučeni, za ovu, ali i buduće nadogradnje (uključujući i one koje provodite svaki dan).

- Logirajte [1] se za dodavanje komentara

pon, 2008-03-31 11:20 - Željko Borš**Vijesti:** Linux [8]
Kuharice: Za sistemce [3]
Kategorije: Servisi [4]
Vote: 0

No votes yet

Apache2: problem s prikazom slova s dijakritičkim znakovima nakon prijelaza s Apacheja 1.X


Mnogi sistem inženjeri su nakon prijelaza s Apache 1.x na Apache 2.x web poslužitelj primjetili da im se na web stranicama ne prikazuju dobro hrvatska slova s dijakritičkim znakovima. Gotovo uvijek je uzrok bila zaostala direktiva u konfiguraciji:

```
AddDefaultCharset ISO-8859-1
```

Direktiva AddDefaultCharset je namijenjena za dodjeljivanje osnovne kodne stranice (*charseta*) vašim web stranicama (kod nas su to najčešće ISO8859-2 i Windows CP-1250). Ukoliko je u konfiguraciji navedena neka druga kodna stranica (kao u ovom primjeru ISO-8859-1), onda ona ima prioritet nad eventualnim META tagovima unutar vaših stranica (iako, zadnju riječ ima korisnikov *browser*). To znači da će stranice biti prikazane u pogrešnoj kodnoj stranici, i zahtijevat će od korisnika da ručno podese svoje *browsere.*

Da ne duljimo, najjednostavnije rješenje problema je jednostavno zakomentirati ovu direktivu u /etc/apache2/apache2.conf, podrazumijevajući da su vam stranice ispravno podešene i imaju navedenu ispravnu kodnu stranicu u META zaglavljima. Ukoliko to ne pomogne, možete postaviti direktivu *AddDefaultCharset* na odgovarajuću kodnu stranicu u konfiguraciji svakog virtualnog hosta koji to zahtijeva, ili je podesiti u datoteci apache2.conf ukoliko vam je cijeli *site* u istoj kodnoj stranici.

Nakon promjena uvijek trebate *reloadati* apache poslužitelj, no to vjerojatno i sami znate:

```
# /etc/init.d/apache2 reload
```

- Logirajte [1] se za dodavanje komentara

čet, 2009-01-01 21:09 - Željko Boroš**Kuharice:** Linux [7]
Kategorije: Servisi [4]
Vote: 5

Vaša ocjena: Nema Average: 5 (1 vote)

Apache2: razlike u odnosu na Apache 1.x

Apache u inačici 2 donosi dosta novosti, ali su promjene većinom dubinske. Najuočljivije su, ipak, one u konfiguraciji. Ni one nisu prevelike, ali su ovoljne da mogu prouzročiti neispravan rad Apache web servisa, odnosno da se apache daemon uopće ne pokrene. Ukoliko na to dodamo Debian specifičnosti, ne treba posebno napominjati da je važno obratiti pozornost na te razlike. Napominjemo da je ovdje (uglavnom) opisana situacija nastala nakon instalacije paketa apache2-cn, koji pokušava prebaciti što više postojeće konfiguracije u novi format.

1. Konfiguracijske datoteke

Konfiguracijske datoteke su razlomljene na više dijelova, po jedna za svaki virtualni poslužitelj. Direktorij u kojem se nalaze sve konfiguracijske datoteke je sada, logično, /etc/apache2. Nekada najvažnija i često jedina konfiguracijska datoteka httpd.conf je sada samo tu radi kompatibilnosti. Nju zamjenjuje datoteka /etc/apache2/apache2.conf, koja sadrži osnovnu konfiguraciju apache poslužitelja i koja vrijedi za sve virtualne poslužitelje. Oduprite se izazovu i ne zlorabite ovu datoteku, nego svaki virtualni web poslužitelj posebno podešavajte (ne treba svakom virtualnom poslužitelju isti set opcija - iz sigurnosnih i drugih razloga uvijek stavite najmanji mogući).

Neke su opcije nekad smještene u monolitni httpd.conf izdvojene u posebne datoteke, npr. portovi na kojima poslužitelj sluša se nalaze u datoteci ports.conf. Svi virtualni poslužitelji imaju vlastite datoteke. O njima malo kasnije.

I dalje postoji conf.d direktorij, u koji možete staviti svoje konfiguracije, što ste pretpostavljamo davno već napravili i s direktivom Include uključivali iz httpd.conf.

```
# ls -l /etc/apache2/conf.d/
-rw-r--r-- 1 root root 657 Mar 19 000-carnet.conf
lrwxrwxrwx 1 root root  29 Mar 28 amavis-stats-cn.conf -> /etc/amavis-
stats/apache.conf
-rw-r--r-- 1 root root 717 Jan 27 apache2-doc
-rw-r--r-- 1 root root 198 Feb 18 mailman-cn.conf
lrwxrwxrwx 1 root root  23 Apr  6 moodle -> /etc/moodle/apache.conf
lrwxrwxrwx 1 root root  27 Mar 31 phpmyadmin.conf -> /etc/phpmyadmin/apache.conf
lrwxrwxrwx 1 root root  29 Feb 20 squirrelmail-
cn.conf -> /etc/squirrelmail/apache.conf
```

2. Lokacija logova

Lokacija logova je, slijedeći logiku imena paketa, /var/log/apache2. No, ukoliko u konfiguraciji vaših virtualnih poslužitelja imate navedeno CustomLog ili ErrorLog u /var/log/apache, oni će tamo i ostati. Opasnost postoji ukoliko potpuno uklonite apache 1, jer će se obrisati i vaš logovi koje se i dalje snimaju u /var/log/apache. Dakle, nakon instalacije apache2-cn, još neko vrijeme nemojte brisati staru konfiguraciju.

3. Virtualni poslužitelji.

Svaki virtualni poslužitelj ima svoju konfiguracijsku datoteku. Sve se one nalaze u /etc/apache2/sites-available u ovom obliku:

```
-rw-------  1 root root  612 Feb 26 08:45 003-site
```

```
-rw-------  1 root root 1167 Aug 16  2007 default
-rw-------  1 root root  449 Feb 26 08:45 moodle.institucija.hr
-rw-------  1 root root  499 Feb 26 08:45 site.institucija.hr
-rw-------  1 root root  324 Feb 26 08:45 www.institucija.hr
```

Ukoliko virtualni web poslužitelj nije bio definiran s imenom, dobit će generičko brojčano ime, no samo ime datoteke nije bitno. Datoteka default određuje ponašanje poslužitelja ukoliko nemate definiran nijedan virtualni poslužitelj, odnosno vrijedi samo ako ste apache2 instalirali na čisti, novi poslužitelj. Ukoliko u njoj ima nešto što vam treba, prebacite u odgovarajući virtualni poslužitelj, ili eventualno u apache2.conf.

Ono što je zanimljivo je način na koji se ovi virtualni poslužitelji uključuju. Naime, nije dovoljno da imate definiran virtualni poslužitelj, nego on mora biti simbolički linkan na datoteku u direktoriju /etc/apache2/sites-enabled. Ukoliko link u direktoriju ne postoji, postojeća konfiguracija neće biti učitana. Ovo omogućuje veliku fleksibilnost i mogućnost eksperimentiranja bez opasnosti da uništite postojeću konfiguraciju: jednostavnom promjenom linka na datoteku dobijate sasvim drugačiju konfiguraciju.

Linkanje se ne mora obavljati ručno, nego za to postoje dvije naredbe: a2ensite i a2dissite. Kako bi uključili novi virtualni poslužitelj, napravite:

```
# a2ensite www.institucija.hr
Site www.institucija.hr installed; run /etc/init.d/apache2 reload to enable.
```

Obrnuto, virtualni poslužitelj se isključuje sa:

```
# a2dissite www.institucija.hr
Site www.institucija.hr disabled; run /etc/init.d/apache2 reload to fully disable
```

Da bi navedene naredbe dobile svoj učinak, morate napraviti po preporuci i restartati apache2.

4. Moduli

Isto kao i virtualni poslužitelji, i moduli su izdvojeni iz zajedničke konfiguracije. Sada se svi moduli nalaze u direktoriju /etc/apache2/mods-available. Pojedine module uključujete naredbom:

```
# a2enmod modul
Module modul installed; run /etc/init.d/apache2 force-reload to enable.
```

Isto tako, pojedine module isključujete sa naredbom:

```
# a2dismod modul
Module modul disabled; run /etc/init.d/apache2 force-reload to fully disable
```

Da bi navedene naredbe dobile svoj učinak, morate napraviti po preporuci i restartati apache2.

- Logirajte [1] se za dodavanje komentara

pet, 2008-04-11 14:33 - Željko Borš**Vijesti:** Linux [8]
Kuharice: Za sistemce [3]
Kategorije: Servisi [4]
Vote: 4.5

Vaša ocjena: Nema Average: 4.5 (2 votes)

Apache: Problem s RADIUS autentikacijom

 U nekoliko članaka na portalu objasnili smo kako da dijelove svog weba zaštitite određenom korisničkom oznakom i zaporkom, uključujući i onu iz AAI@EduHr sustava (članak "Autentikacija Apache web poslužitelja putem LDAP-a i RADIUS-a [6]" i srodan "Autentikacija osnovnih servisa putem LDAP-a [9]"). No, u kombinaciji s drugim metodama autentikacije može doći do određenih problema.

Naime, kod uporabe mod_auth_radius modula potrebno je uključiti direktivu

```
AuthRadiusActive On
```

koja za posljedicu ima, osim što uključuje RADIUS autentikaciju, da pokušava autenticirati korisnike prije drugih oblika autentikacije. Ovo znači da ako imate uključenu autentikaciju preko .htpasswd datoteke, ona jednostavno neće raditi. Da je nešto sumnjivo, može se vidjeti u error.logu:

```
[error] (2)No such file or directory: access to /secure/location failed
for 192.168.1.1, reason: RADIUS authentication failed for user korisnik
```

Rješenje je, nakon čitanja dokumentacije na http://www.freeradius.org/mod_auth_radius/httpd.conf [10] vrlo jednostavno. Za svaku lokaciju, odnosno direktorij (određene sa <Location> i <Directory>) u kojima ne trebate RADIUS autentikaciju morate isključiti direktivu AuthRadiusActive:

```
AuthRadiusActive Off
```

Nakon reloada apache poslužitelja, dijelovi weba koji se ne autenticiraju preko RADIUS protokola, radit će kako je to i predviđeno, preko .htpasswd baze korisnika i njihovih zaporki.

- Logirajte [1] se za dodavanje komentara

sri, 2007-12-05 14:52 - Željko Borš**Vijesti:** Linux [8]
Kuharice: Za sistemce [3]
Kategorije: Servisi [4]

Vote: 0

No votes yet

Apache: SSL ne podržava višestruke virtualne hostove

Apache je jedan od najpopularnijih softvera za web servise, a taj status obično dolazi od činjenice da se takav softver razlikuje od drugih po tome što ima najviše mogućnosti, najfleksibilniji je i slično. U slučaju takvog softvera korisnici imaju dojam da je svemoguć, što naravno zna razočarati kad se dozna da to nije slučaj.

U slučaju Apacheja, to se najviše može vidjeti u činjenici da nije moguće imati više SSL virtualnih hostova, odnosno nije moguće preko HTTPS protokola pristupiti na nekoliko različitih adresa na istom poslužitelju.

Dakle, nije moguće istovremeno imati SSL adrese

https://nesto.institucija.hr

i

https://www.institucija.hr

Ovo, sasvim ozbiljno ograničenje, se može zaobići na nekoliko načina, no nažalost nijedan nije previše praktičan, no nije nemoguće za riješiti. Naravno, potrebno je nešto više iskustva u radu s nekim servisima i samim operativnim sustavom.

Razlog zašto je nemoguće imati više adresa, odnosno virtualnih hostova, leži u činjenici što se SSL nalazi na različitom mrežnom sloju od HTTP protokola, te ga enkapsulira unutar sebe. Ovo znači da nema "Host:" zaglavlja po kojemu se razlikuju virtualni hostovi, te nema mogućnosti da Apache zna o kojem se virtualnom hostu radi (obično se svodi na prvi definirani).

Postoji nekoliko načina rješavanja ovog problema.

1. Jedan SSL poslužitelj za sve potrebe

Ukoliko želite nekoliko servisa osigurati preko SSL-a, možete ih jednostavno staviti unutar nekog dediciranog virtualnog hosta, primjerice secure.institicija.hr:

https://secure.institucija.hr/webmail

ili

https://secure.institucija.hr/prijava

Ovo je najlakše za ostvariti, jer posebnog podešavanja i nema, potrebno je samo definirati jedan

VHOST sa SSL-om, te u njegov DOCUMENT_ROOT stavljati webove koje želite zaštititi (ili ih symlinkati s neke druge lokacije). Tipičan izgled SSL VHOST-a, primjerice /etc/apache2/sites-available/ssl:

```
<IfModule mod_ssl.c>

# Since SSL has no NameVirtualHosts, and we don't support machines with
# multiple IP addresses yet, make this a simple default config.

<VirtualHost _default_:443>
    ServerAdmin admin@institucija.hr
    ServerName server.institucija.hr
    DocumentRoot /var/www/secure
    LogLevel warn
    ErrorLog /var/log/apache2/ssl-secure.institucija.hr-error.log
    CustomLog /var/log/apache2/ssl-secure.institucija.hr-access.log combined

    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/secure.institucija.hr.crt
    SSLCertificateKeyFile /etc/ssl/private/secure.institucija.hr.key
    SSLCertificateChainFile /etc/ssl/certs/secure.institucija.hr.chain
    # Needed for older MSIE6 patch levels
    SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown

</VirtualHost>

</IfModule>
```

Kod definiranja ostalih VHOST-ova (koji nisu SSL), rabite ovaj oblik VirtualHost direktive:

```
NameVirtualHost 161.53.XXX.YYY:80
```

Druga dva načina su definiranje joše jedne adrese na mrežnom sučelju, ili stavljanje druge mrežne kartice u poslužitelj, te definiranje SSL VHOST-a na nekom drugom portu (default je 443).

2. druga IP adresa

Ako dodamo dodatnu mrežnu karticu na poslužitelj, ili kreiramo virtualno mrežno sučelje, možemo na novim IP adresama poslužitelja definirati drugi virtualni SSL host, zadržavajući *defaultni* port 443. Ipak, ukoliko vama niti vašim korisnicima nije problem koristiti drugi SSL port, lakši način je definirati SSL VHOST na drugom portu.

3. drugi port

Moguće je definirati virtualni host na drugom portu, ali tada je potrebno navoditi taj port u adresi. Tako primjerice umjesto jednostavnog:

https://webmail.institucija.hr

moramo navesti

https://webmail.institucija.hr:81

Port je naravno, proizvoljan, no pazite da ne uzmete neki koji je već u uporabi.

Za svaki SSL VHOST koji želite definirati stavite:

```
NameVirtualHost 161.53.XXX.ZZZ:81

<VirtualHost 161.53.XXX.ZZZ:81>
  ServerName some.domain.com
  # SSL i druge opcije
</VirtualHost>
```

Ipak, moramo reći da rješenje za više SSL hostova na jednoj adresi postoji, i to u vidu dodatnih modula, primjerice mod_gnutls [11]. Ovaj modul podržava SNI (Server Name Identification [12]) proširenja TLS-a, no još se vodi kao eksperimentalan, pa ga ne možemo preporučiti za uporabu na produkcijskom poslužitelju. Ukoliko ipak želite probati, pogledajte upute na http://www.der-eremit.de/post/13589628448/ssl-enabled-name-based-virtual-hosts-with-mod-gnutls [13].

Dodatno, ukoliko ste napravili certifikat s više FQDN-ova (ili wildcard certifikat, koji nije preporučljiv iz sigurnosnih razloga), *vjerojatno* ćete moći ostvariti da imate više SSL VHOST-ova. Kažemo "vjerojatno" jer neki su webmasteri prijavili da im ovakvo rješenje jednostavno ne radi. Nadalje, ukoliko niste naveli sve VHOST-ove u alt_names sekciji zahtjeva za certifikatom, morat ćete ponoviti cijelu proceduru i možda čak povući (*revokati*) stari certifikat. Na kraju, sve popularniji uređaji za surfanje - mobilni uređaji - najčešće ne podržavaju SNI, što bi nekima moglo biti dosta važno.

Nove inačice OpenSSL-a također obećavaju podršku za SNI (od inačice 0.9.8j koja nije u Lenny distribuciji), pa će problem višestrukih VHOST-ova na jednoj IP adresi vjerojatno biti trajno riješen.

- Logirajte [1] se za dodavanje komentara

čet, 2010-06-24 13:16 - Željko Boroš**Kuharice:** Linux [7]
Kategorije: Servisi [4]
Vote: 0

No votes yet

Apache: Tilda u web adresi

Nedavno je na listi sistemci postavljeno pitanje kako, umjesto klasičnog načina zapisivanja korisničkih stranica pomoću tilde http://www.ustanova.hr/~korisnik/, doći do adrese oblika http://korisnik.ustanova.hr/.

Jedan od odgovora uključivao je RewriteRules i potakao me da pokušam ponuditi jedno od mogućih rješenja problema.

Za početak, potrebno je u DNS-u postaviti DNS wildcard za poddomenu *.ustanova.hr:

```
*.ustanova.hr.          IN      CNAME           imeposluzitelja
```

Zatim slijedi preinaka u datoteci httpd.conf:

```
<VirtualHost 192.168.1.1:80>
    ServerName korisnici.ustanova.hr
    ServerAlias *.ustanova.hr
    DocumentRoot /var/www/korisnici/

    UseCanonicalName Off

    RewriteEngine on
    RewriteLogLevel 1
    RewriteLog /var/log/apache/rewrite.log

    RewriteCond %{HTTP_host} ^([a-z0-9][-a-z0-9]+)\.ustanova\.hr\.?(:80)?$ [NC]
    RewriteCond /home/%1/public_html -d
    RewriteRule ^(.*) /home/%1/public_html/$1 [L]
</VirtualHost>
```

Malo o konfiguraciji liniju po liniju.

Pretpostavka je da u DNS-u imamo zapis korisnici, na adresi 192.168.1.1, koji na poslužitelju gleda u direktorij /var/www/korisnici. Ovdje je to urađeno samo zbog primjera uz pretpostavku da je www.ustanova.hr izdvojen kao zasebni virtualni poslužitelj.

ServerAlias direktiva postavlja alternativno ime poslužitelja na wildcard vrijednost, a direktiva „UseCanonicalName Off“ kaže poslužitelju da za ime koristi ono što mu klijent pošalje (u našem slučaju korisnik.ustanova.hr).

Slijede minimalne postavke za RewriteEngine. RewriteLogLevel može imati vrijednost od 0 do 9, pri čemu je 0-isključeno, a 9 zapisuje svaku akciju. Iako je viša vrijednost dobra za debugiranje i kontrolu rada, u radu se, zbog procesorskog i diskovnog opterećenja, ne preporuča vrijednost veća od 1.

Nakon toga slijede RewriteCond – pravila koja moraju biti zadovoljena kako bi uslijedio RewriteRule. Ovdje se provjerava da li je klijent poslao HTTP_HOST oblika nesto.ustanova.hr te da li u tom slučaju postoji direktorij /home/nesto/public_html. Možda je ovdje dobro napomenuti kako smo se ovako ograničili na klijente koji podržavaju HTTP1/1.

Ukoliko su RewriteCond zadovoljeni, izvršava se RewriteRule koji radi internu redirekciju adrese korisnik.ustanova.hr na direktorij /home/korisnik/public_html. Naravno, ukoliko su korisnički direktoriji negdje drugdje (npr. /home/studenti/korisnik/public_html), potrebno je prilagoditi odgovarajuću putanju do njih.

Ono što bi svakako valjalo napomenuti je da bi, u slučaju implementacije ovakvog ili sličnog rješenja, trebalo paziti na zagađenje imeničkog prostora ustanove. U tom smislu bi možda trebalo razmisliti o uporabi adresa oblika korisnik.nesto.ustanova.hr (npr. pero.web.ustanova.hr ili pero.korisnici.ustanova.hr).

- Logirajte [1] se za dodavanje komentara

uto, 2007-07-17 15:53 - Ljubomir Hrboka**Kuharice:** Za sistemce [3]

Kategorije: Servisi [4]
Vote: 0

No votes yet

Apache: kako zaštiti pojedine dijelove weba preko access liste?

 Iako je web javni servis, ponekad želimo da samo određena skupina ljudi može vidjeti informacije na njemu, primjerice samo zaposlenici na instituciji.

Mada je bolje da interne informacije budu na posebnom poslužitelju unutar lokalne mreže, možemo se poslužiti i mogućnostima Apache modula mod_access. Modul, naravno, mora biti uključen u /etc/apache/modules.conf:

```
LoadModule access_module /usr/lib/apache/1.3/mod_access.so
```

Slijedeća konfiguracija može biti navedena u /etc/apache/httpd.conf, ali je bolje da se nalazi u posebnoj datoteci u /etc/apache/conf.d/ direktoriju, primjerice local.conf. Datoteke u tom direktoriju se automatski uključuju u konfiguraciju preko Include direktive koja se nalazi u datoteci /etc/apache/httpd.conf.

```
<Directory /var/www/private>
Order Deny,Allow
Deny from All
Allow from 192.168.1 192.168.2 161.53.xxx
</Directory>
```

Zabrana pristupa djeluje samo u direktoriju /var/www/private. Direktiva Order određuje redoslijed evaluacije "Deny" i "Allow" direktiva. "Order Deny,Allow" (logično) znači da se prvo evaluiraju direktive upisane pod Deny, a tek onda pod Allow. Ključna riječ "All" označava **sve** IP adrese. Logika evaluiranja je sljedeća:

- ako je IP adresa klijenta upisana pod Deny, zabranjuje mu se pristup **osim** ako je naveden i pod Allow. Ovaj način **nije u skladu** s uobičajenim načinom konfiguriranja (primjerice firewalla), stoga tu treba biti oprezan. Ukoliko IP adresa nije navedena nigdje, klijentu se omogućava pristup.

Da je bilo navedeno "Order Allow,Deny", logika bi bila sljedeća:

- ako je IP adresa klijenta navedena pod Allow, pristup mu se dopušta, ali samo u slučaju da se IP adresa **ne nalazi** i u Deny. Da je bila navedena samo pod Deny, pristup bi mu bio zabranjen. Ako IP adresa nije navedena ni pod Allow, ni pod Deny, pristup bi mu bio zabranjen, kao i u slučaju da je IP adresa navedena i pod Allow i pod Deny.

U ovom konkretnom slučaju, zamjena redoslijeda bi zapravo značila **zabranu** pristupa s navedenih IP adresa. Uvijek se evaluiraju obje strane prije nego se pristup dopusti ili zabrani, i uvijek "pobjeđuje" zadnja navedena direktiva u Orderu. Dakle, oprez i testiranje je nužno.

Vrijedi napomenuti da nije nužno navesti IP adresu, to može biti i domena:

```
 Order Deny,Allow
Deny from all
Allow from domena.hr
```

Više informacija, kao i uvijek, možete naći na webu:

http://httpd.apache.org/docs/1.3/mod/mod_access.html [14]

- Logirajte [1] se za dodavanje komentara

pet, 2007-05-04 15:03 - Željko Borš**Kuharice:** Za sistemce [3]
Kategorije: Servisi [4]
Vote: 0

No votes yet

Autorizacija Apache web poslužitelja putem LDAP-a i RADIUS-a

U prošlomjese?nom ?lanku Autentikacija osnovnih servisa putem LDAP-a **[5]** objasnili smo autorizaciju osnovnih servisa putem LDAP-a i RADIUS-a. Sada ?emo pokazati autorizaciju Apache web poslužitelja putem istih autentikacijskih i autorizacijskih mehanizama.

Apache web poslužitelj može koristiti autorizaciju korisnika preko RADIUS-a. Za to se koristi modul mod_auth_radius. Taj modul se nalazi u Debian paketu libapache-mod-auth-radius. Zato ga najprije instalirajmo:

```
# apt-get install libapache-mod-auth-radius
```

Zatim treba u konfiguraciju FreeRADIUS-a prijaviti tog poslužitelja kao klijenta:

```
# U nasem slucaju je klijent localhost
client 127.0.0.1 {

        secret = neki_secret
        shortname = localhost

}
```

Ovaj secret treba prenijeti u konfiguraciju mod_auth_radius modula (kod nas je u /etc/apache/conf.d/mod-auth-radius.conf). Prilikom svake izmjene konfiguracijskih datoteka FreeRADIUS-a, potrebno ga je restartati:

```
# /etc/init.d/freeradius restart
```

Zatim u datoteku /etc/apache/conf.d/mod-auth-radius.conf upišemo sljede?e:

```
<IfModule mod_auth_radius.c>

        AddRadiusAuth localhost:1812 neki_secret 5:3
        AddRadiusCookieValid 5

</IfModule>
```

neki_secret stavimo proizvoljno, i dobro je da je tajni. Brojevi 5:3 zna?e da je timeout 5 sekundi, te da je broj pokušaja 3, ali se ove vrijednosti mogu proizvoljno mijenjati.

```
<Location /ono-ste-zelimo-zastititi/>

        AuthType BasicAuthName "RADIUS authentication for localhost"
        AuthAuthoritative off
        AuthRadiusAuthoritative on
        AuthRadiusCookieValid 5
        AuthRadiusActive On
```

```
        require valid-user

</Location>
```

Naravno, Location se mijenja ovisno o tome koji dio weba želimo zaštititi lozinkom.

Nakon svake izmjene datoteka unutar /etc/apache/conf.d/ direktorija treba restartati Apache:

```
# /etc/init.d/apache restart
```

Web stranici na adresi **http://www.ustanova.hr/ono-sto-zelimo-zastiti/** može se pristupiti samo korisni?kim imenom i lozinkom iz lokalnog LDAP-a, koriste?i lokalni RADIUS.

Postoji i drugi na?in. Dovoljno je u /etc/apache/conf.d/mod-auth-radius.conf staviti:

```
<IfModule mod_auth_radius.c>

        AddRadiusAuth localhost:1812 neki_secret 5:3

        AddRadiusCookieValid 5

</IfModule>
```

Direktorij koji želimo zaštiti upišemo u datoteku .htaccess:

```
AuthType Basic

AuthName "RADIUS authentication for localhost"

AuthAuthoritative off

AuthRadiusAuthoritative on
```

```
AuthRadiusCookieValid 5

AuthRadiusActive On



require valid-user
```

I to je to!

- Logirajte [1] se za dodavanje komentara

pon, 2006-02-27 13:24 - Uredništvo**Kuharice:** Za sistemce [3]
Kategorije: Servisi [4]
Vote: 4

Vaša ocjena: Nema Average: 4 (1 vote)

Kako forsirati SSL samo na određenim dijelovima web stranica?

U ovom članku opet ćemo se baviti web poslužiteljem Apache i SSL-om. Jedan od često postavljanih, a dosad nepokrivenih pitanja je kako forsirati SSL na određenom dijelu weba, ili kako preusmjeriti ne-SSL zahtjeve na neki drugi SSL VHOST.

Podrazumijevamo da je sve što se tiče SSL-a ispravno podešeno (certifikat je ispravan, poželjno je da je potpisan od strane nekog CA, SSL je uključen na VHOST-u i slično), te da se vašim stranicama može pristupati preko prefiksa **https://**.

Za preusmjeravanje možemo rabiti redirekciju, ili upotrijebiti popularne Redirect direktive. Uzet ćemo gotovi primjer, iz našeg paketa squirrelmail-cn, pa ćemo imati primjer iz zaista svima poznatog i dostupnog paketa.

```
# Force SSL for /webmail -> you can still use /squirrelmail
Alias /webmail /usr/share/squirrelmail

<IfModule mod_rewrite.c>
  <IfModule mod_ssl.c>
    <Location /webmail>
      RewriteEngine on
```

```
      RewriteCond %{HTTPS} !=on
      RewriteRule . https://%{HTTP_HOST}%{REQUEST_URI}  [L]
    </Location>
  </IfModule>
</IfModule>
```

S naredbom "**RewriteEngine on**" uključujemo Rewrite modul, dok sa **RewriteCond** provjeravamo je li SSL uključen. Ukoliko SSL nije uključen, a pokušamo rabiti Rewrite direktive, dobit ćemo poruku o grešci i web stranice neće raditi.

Glavni posao obavlja direktiva **RewriteRule**: ona će preusmjeriti bilo koji zahtjev (".") na https://, a samu adresu i URL će složiti iz originalnog zahtjeva. Na kraju je zastavica ("flag") "[L]", koja znači jednostavno "Last" i zaustavlja procesiranje eventualnih dodatnih RewriteRule direktiva. U suprotnom, u određenim situacijama bi moglo doći do beskonačne petlje u procesiranju Rewrite direktiva.

Ovaj se primjer nalazi u datoteci /etc/squirrelmail/apache.conf, koja je simbolički povezana u konfiguracijskom direktoriju apacheja:

```
# ls -l /etc/apache2/conf.d/squirrelmail-cn.conf
lrwxrwxrwx 1 root root 29 Feb 20  2008
/etc/apache2/conf.d/squirrelmail-cn.conf -> /etc/squirrelmail/apache.conf
```

Ovo navodimo jer će na ovaj način redirekcija, odnosno prepisivanje URL-ova preko Rewrite direktiva biti dostupna u svim VHOST-ovima, pa pripazite da navedeni VHOST ima uključenu podršku za SSL.

Na sličan način vi možete preusmjeriti dio weba na sigurnu SSL konekciju, samo stavite konfiguraciju unutar direktorija /etc/apache2/conf.d/, ili simbolički povežite datoteku koja se zapravo nalazi na nekom drugom mjestu:

```
# ln -s /etc/mojprogram/apache.conf /etc/apache2/conf.d/mojprogram.conf
```

Na kraju obavezno napravite reload ili restartajte apache:

```
# /etc/init.d/apache reload
```

Sad će svaki poziv stranici http://www.domena.hr/webmail biti preusmjeren na **https://** inačicu iste te stranice. Ostale stranice će raditi uobičajeno (obratite pažnju na direktivu "Location" koja određuje ovakvo ponašanje, da ne biste obuhvatili više vašeg weba nego što treba).

Još dosta trikova oko preusmjeravanja sa ne-SSL na SSL stranice ili druge VHOST-ove potražite na adresi:

http://www.askapache.com/htaccess/ssl-example-usage-in-htaccess.html [15]

- Logirajte [1] se za dodavanje komentara

uto, 2010-06-29 13:12 - Željko Borš**Vote:** 5

Vaša ocjena: Nema Average: 5 (2 votes)

Source URL: https://sysportal.carnet.hr/node/198

Links
[1] https://sysportal.carnet.hr/sysportallogin
[2] http://httpd.apache.org/docs/2.2/vhosts/
[3] https://sysportal.carnet.hr/taxonomy/term/22
[4] https://sysportal.carnet.hr/taxonomy/term/28
[5] https://sysportal.carnet.hr/node/42
[6] https://sysportal.carnet.hr/node/41
[7] https://sysportal.carnet.hr/taxonomy/term/17
[8] https://sysportal.carnet.hr/taxonomy/term/11
[9] http://sistemac.carnet.hr/node/42
[10] http://www.freeradius.org/mod_auth_radius/httpd.conf
[11] http://www.outoforder.cc/projects/apache/mod_gnutls/
[12] http://en.wikipedia.org/wiki/Server_Name_Indication
[13] http://www.der-eremit.de/post/13589628448/ssl-enabled-name-based-virtual-hosts-with-mod-gnutls
[14] http://httpd.apache.org/docs/1.3/mod/mod_access.html
[15] http://www.askapache.com/htaccess/ssl-example-usage-in-htaccess.html