

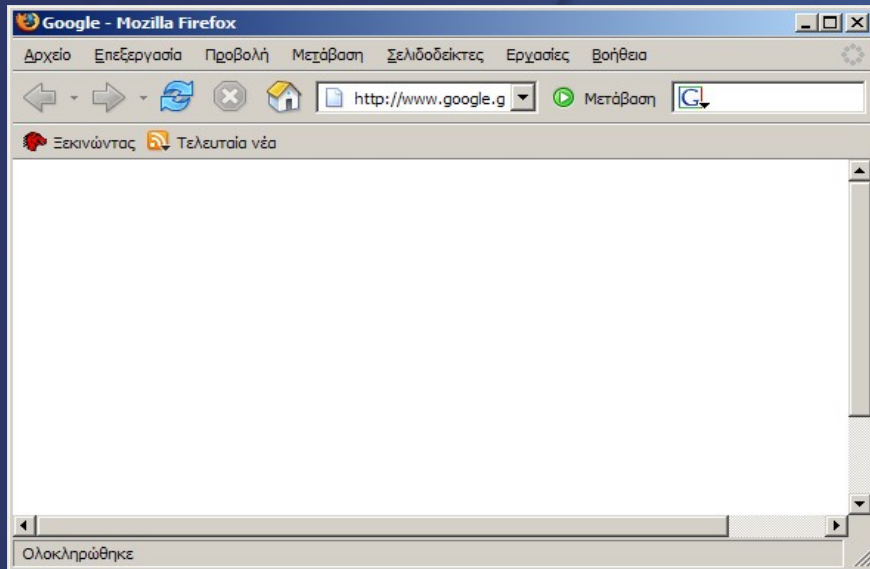
Setting up Apache with PHP and MySQL

Χαρίτων Καραμήτας <chariton.karamitas@echothrust.com>
Παναγιώτης Ευστρατίου <panagiotis.efstratiou@echothrust.com>

Τι είναι τα services ;

- Πρόκειται για εφαρμογές που τρέχουν σε κάποιους υπολογιστές και περιμένουν συνδέσεις από άλλους υπολογιστές με σκοπό την παροχή κάποιων υπηρεσιών.
- Η παροχή αυτών των υπηρεσιών γίνεται ακολουθώντας κάποιους κανόνες, ή πιο σωστά, **πρωτόκολλα**.
- Ο Apache είναι μια υπηρεσία (service) που “μιλάει” το πρωτόκολλο **HTTP**.

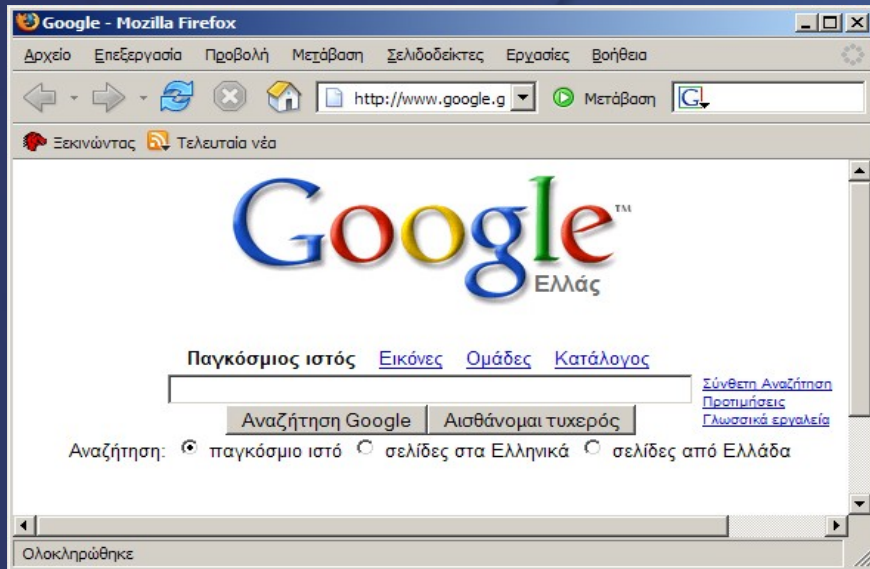
Διαδικασία πλοήγησης σε σελίδες στο internet



- Ο χρήστης ανοίγει τον αγαπημένο του browser, Mozilla ή Internet Explorer.
- Ο χρήστης εισάγει μια διεύθυνση π.χ **www.google.com**.
- Ο υπολογιστής με αυτή τη διεύθυνση τρέχει Apache και κάνει δεκτή τη σύνδεση.
- Ο υπολογιστής του χρήστη και ο **www.google.com** συνομιλούν με το πρωτόκολλο HTTP.
- Το αποτέλεσμα είναι να εμφανίζεται η γνωστή σελίδα του google στον χρήστη.

Διαδικασία πλοήγησης σε σελίδες στο internet

- Ο χρήστης ανοίγει τον αγαπημένο του browser, Mozilla ή Internet Explorer.
- Ο χρήστης εισάγει μια διεύθυνση π.χ **www.google.com**.
- Ο υπολογιστής με αυτή τη διεύθυνση τρέχει Apache και κάνει δεκτή τη σύνδεση.
- Ο υπολογιστής του χρήστη και ο **www.google.com** συνομιλούν με το πρωτοκολλο HTTP.
- Το αποτέλεσμα είναι να εμφανίζεται η γνωστή σελίδα του google στον χρήστη.



Τι ακριβώς θα χρειαστούμε ;

- Apache 1.3.x
Πολύ γνωστός HTTP server. Δηλαδή υπηρεσία που “μιλάει” το πρωτόκολλο HTTP (www.apache.org).
- OpenSSL και ModSSL
Επιπρόσθετο κομμάτι του Apache για ασφαλείς συνδέσεις (www.openssl.org και www.modssl.org).
- PHP
Γλώσσα προγραμματισμού για δυναμικό περιεχόμενο στον Apache (www.php.net και www.hardened-php.net).
- MySQL
Γνωστή βάση δεδομένων (www.mysql.com).

Τι είναι το SSL ;

- Netscape's SSL (**Secure Sockets Layer**).
- Τελευταία έκδοση είναι το SSLv3.
- Η ασφάλεια του SSL βασίζεται σε 3 βασικά χαρακτηριστικά:

Ισχυρή κρυπτογράφηση 2 επιπέδων (asymmetric & symmetric cryptography e.g **RSA** και **RC4**).

Ακεραιότητα δεδομένων (One way hash encryption e.g **MD5**).

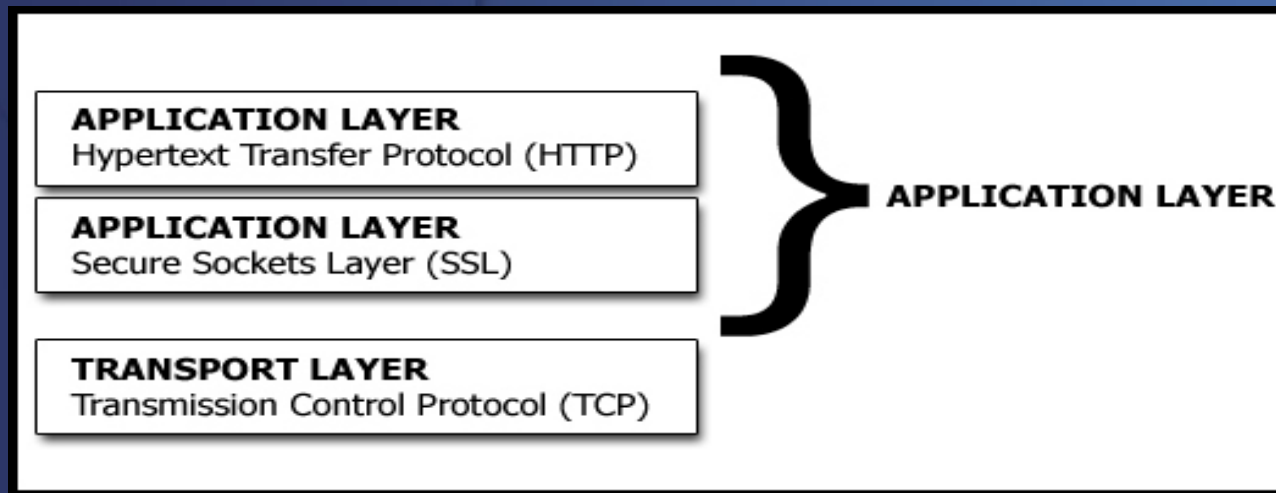
Εμπιστοσύνη (**SSL x509 certificates**).

<http://wp.netscape.com/eng/ssl3/draft302.txt>

http://en.wikipedia.org/wiki/Secure_Sockets_Layer

Τι είναι το SSL ;

- Το SSL λειτουργεί στο Application Layer του OSI:



- Λειτουργεί συνήθως πάνω από το **TCP**.
- Το HTTP μαζί με SSL συχνά αναφέρεται και ως **HTTPS** (**Hypertext Transfer Protocol - Secure**).

Πόσο ασφαλές είναι το SSL ;

- Η μόνη επίθεση που στοχεύει το πρωτόκολλο και μπορεί να είναι επιτυχής είναι το **SSL MiM (Man In the Middle)**.

- Μία επιτυχημένη επίθεση:

Προϋποθέτει την ύπαρξη κάποιων συνθηκών.

Συνδιάζει αδυναμίες και σε άλλα πρωτόκολλα (e.g **DNS**).

Συνδιάζει αδυναμίες και σε διάφορους browsers (e.g Mozilla).

- Στοχεύει το 3ο βασικό χαρακτηριστικό του SSL, την εμπιστοσύνη μέσω certificates.

Εγκατάσταση – Pre install

- Αρχικά πρέπει να δημιουργήσουμε χρήστες για τον Apache και τη MySQL.

```
$ su -c "useradd -m -d /home/httpd -p 'x' -g nobody httpd"
```

```
$ su -c "useradd -m -d /home/mysql -p 'x' -g nobody mysql"
```

- Έπειτα αναβαθμίζουμε την έκδοση του OpenSSL για να αποφύγουμε κάποια προβλήματα.

```
$ tar -zxf openssl-xxx.tar.gz
```

```
$ cd openssl-xxx
```

```
$ export CFLAGS="-O2"
```

```
$ ./config --prefix=/usr shared
```

```
$ make && make test
```

```
$ su -c "make install"
```

Εγκατάσταση - Install

- Κάνουμε compile τον Apache μαζί με το ModSSL.

```
$ export CFLAGS="-O2"
$ tar -zxf mod_ssl-xxx.tar.gz
$ tar -zxf apache_xxx.tar.gz
$ cd mod_ssl-xxx
$ ./configure --prefix=/usr --with-apache=../apache_xxx
$ cd ../apache_xxx
$ ./configure --prefix=/usr --enable-module=so \
               --enable-module=ssl
$ make
$ su -c "make install"
```

Εγκατάσταση - Install

- Κάνουμε compile τη MySQL. Η εγκατάσταση της PHP απαιτεί να υπάρχει ήδη εγκατεστημένη η MySQL.

```
$ export CFLAGS="-O2"
```

```
$ tar -zxf mysql-xxx.tar.gz
```

```
$ cd mysql-xxx
```

```
$ ./configure --prefix=/usr --enable-thread-safe-client \  
              --enable-assembly --with-extra-charsets=none \  
              --with-pthread --without-pstack \  
              --without-debug --with-embedded-server \  
              --with-mysqld-user=mysql
```

```
$ make
```

```
$ su -c "make install"
```

Εγκατάσταση - Install

- Κάνουμε compile την PHP.

```
$ export CFLAGS="-O2"
```

```
$ tar -jxf php-xxx.tar.bz2
```

```
$ gzcat hardening-patch-xxx.gz | patch -p0
```

```
$ cd php-xxx
```

```
$ ./configure --prefix=/usr --with-mysqli=/usr/bin/mysqli_config \  
--with-mysql=/usr --with-apxs=/usr/bin/apxs \  
--with-curlwrappers --with-curl=/usr \  
--enable-memory-limit --enable-ftp --enable-sockets
```

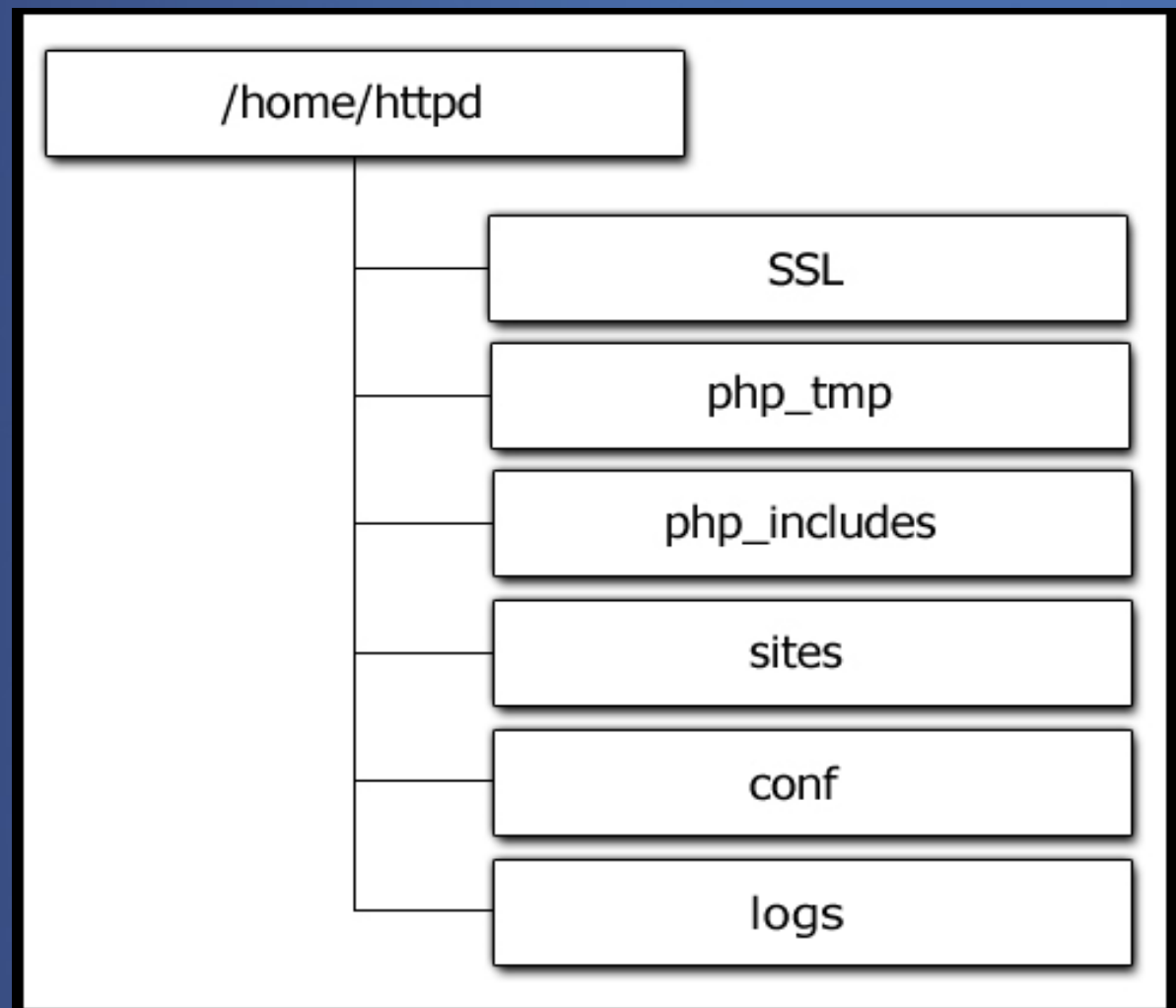
```
$ make
```

```
$ su -c "make install"
```

Εγκατάσταση – Post install

- Δημιουργούμε τους καταλόγους που χρειάζεται ο Apache.

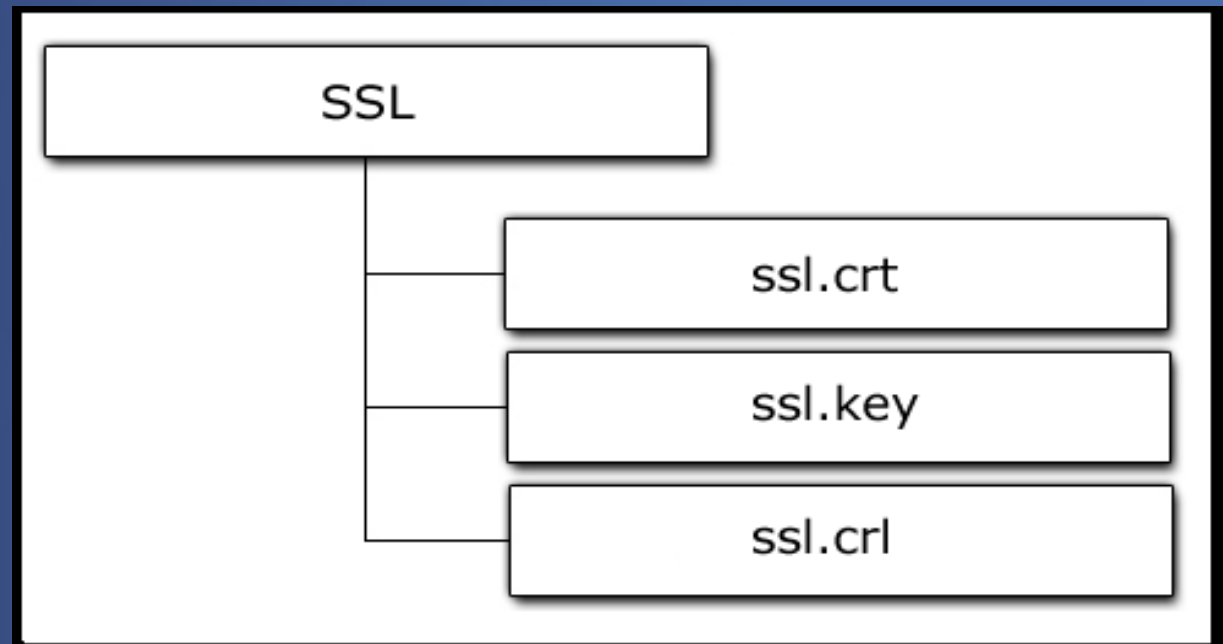
```
$ su httpd  
$ cd /home/httpd  
$ mkdir SSL  
$ mkdir php_tmp  
$ mkdir php_includes  
$ mkdir sites  
$ mkdir conf  
$ mkdir logs
```



Εγκατάσταση – Post install

- Δημιουργούμε τους καταλόγους που χρειάζεται ο Apache.

```
$ cd SSL  
$ mkdir ssl.crt  
$ mkdir ssl.key  
$ mkdir ssl.crl
```



Εγκατάσταση – Post install

- Certificate authority.

```
# openssl req -x509 -days 365 -newkey rsa:1024 -keyout \  
/home/httpd/SSL/ssl.key/ca.key -out /home/httpd/SSL/ssl.crt/ca.crt
```

- Server SSL certificate.

```
# openssl genrsa -out /home/httpd/SSL/ssl.key/server.key 1024  
# openssl req -new -utf8 -out /home/SSL/ssl.crt/server.csr -key \  
/home/httpd/SSL/ssl.key/server.key -days 365  
# openssl x509 -req -in /home/httpd/SSL/ssl.crt/server.csr \  
-out /home/httpd/SSL/ssl.crt/server.crt \  
-CA /home/httpd/SSL/ssl.crt/ca.crt \  
-CAkey /home/httpd/SSL/ssl.key/ca.key -CAcreateserial -sha1
```

Εγκατάσταση – Post install

- Δημιουργία χρηστών για τον Apache.

```
# htpasswd -c <-m|-s|-d> /home/httpd/.apache_pass username_1
```

```
...
```

```
...
```

```
# htpasswd <-m|-s|-d> /home/httpd/.apache_pass username_n
```


Εγκατάσταση – Post install

- Δημιουργούμε τα configuration αρχεία που χρειάζεται ο Apache.

```
$ touch /home/httpd/conf/httpd.conf  
$ touch /home/httpd/conf/mime.types
```

- Αλλάζουμε τα permissions.

```
$ chown -R httpd:nobody /home/httpd/  
$ chmod -R 0700 /home/httpd/  
$ su "chown -R root:root /home/httpd/SSL /home/httpd/conf"  
$ su "chmod -R 0700 /home/httpd/SSL /home/httpd/conf"
```

Εγκατάσταση – Post install

- Απαραίτητες ενέργειες για τη λειτουργία της MySQL.

```
$ su root
# mkdir /var/mysql
# mysql_install_db --user=mysql --datadir=/var/mysql
# chown -R mysql:nobody /var/mysql
# chmod -R 0700 /var/mysql
# ( mysqld_safe --user=mysql --datadir=/var/mysql --secure-auth \
--skip-networking --log-error=/var/log/mysql.err & )
# mysqladmin -u root -p -h localhost password \
    \#s0000p3rp4ssw0rd\!
```

Apache configuration (httpd.conf)

- Ρυθμίσεις για καταγραφή συμβάντων (Logs).

```
CustomLog /home/httpd/logs/apache_details.log "%h %l %u %t  
\"%r\" %>s %b \"%{Referer}i\" \"%{User-agent}i\""
```

```
CustomLog /home/httpd/logs/apache_basic.log "%h %l %u %t  
\"%r\" %>s %v"
```

```
CoreDumpDirectory /home/httpd
```

```
ErrorLog /home/httpd/logs/error.log
```

- Δημιουργούμε 2 αρχεία για την καταγραφή συμβάντων – 1 με βασικές πληροφορίες και ένα με λεπτομέριες.

Apache configuration (httpd.conf)

- Βασικές ρυθμίσεις ασφάλειας.

User httpd
Group nobody

```
# Disabled server options.  
# - Link handling  
# - Server side includes  
# - Directory indexes  
Options -FollowSymLinks  
Options -Includes  
Options -Indexes
```

Apache configuration (httpd.conf)

- Ρυθμίσεις εισερχόμενων συνδέσεων.

Listen 0.0.0.0:80

Listen 0.0.0.0:443

ListenBacklog 10

- Γενικές ρυθμίσεις.

ServerType standalone

ServerRoot /home/httpd

ServerSignature Off

ServerTokens ProductOnly

ServerAdmin httpd@localhost

ServerName localhost

Apache configuration (httpd.conf)

- Ενεργοποίηση της PHP.

```
LoadModule php5_module /usr/libexec/libphp5.so
```

```
AddModule mod_php5.c
```

```
AddType application/x-httpd-php .php .php3 .php5 .phtml .inc .g
```

```
AddType application/x-httpd-php-source .phps
```

```
AddType text/html .htm .html .xhtml
```

Apache configuration (httpd.conf)

- Apache virtual hosts. Παράδειγμα:

```
NameVirtualHost *:80
NameVirtualHost *:443
```

```
<VirtualHost *:80>
    ServerName localhost

    DocumentRoot /home/httpd/sites/localhost
    DirectoryIndex index.html

    # Protect against user enumeration.
    RedirectMatch 400 ~[^/]*
</VirtualHost>
```

Apache configuration (httpd.conf)

- SSL configuration.

SSLCertificateFile /home/httpd/SSL/ssl.crt/server.crt

SSLCertificateKeyFile /home/httpd/SSL/ssl.key/server.key

SSLCertificateChainFile /home/httpd/SSL/ssl.crt/ca.crt

SSLCARevocationPath /home/httpd/SSL/ssl.crl

SSLVerifyClient none

SSLVerifyDepth 0

Apache configuration (httpd.conf)

- SSL virtual host.

```
<VirtualHost _default_:443>
    SSLEngine on
    ServerName localhost
    DocumentRoot /home/httpd/sites/localhost_ssl
    RedirectMatch 400 ~[^/]*
    <Directory /home/httpd/sites/localhost_ssl>
        SSLRequireSSL
        DirectoryIndex index.html
        AllowOverride None
        Options None
        AuthName "Authentication"
        AuthType Basic
        AuthUserFile /home/httpd/.apache_pass
        Require user username_1 ... username_n
    </Directory>
</VirtualHost>
```

Apache configuration (httpd.conf)

- Άρνηση πρόσβασης σε κρυφά αρχεία και φακέλους.

```
<Files ~ "^\. ">  
    Deny from all  
    AllowOverride None  
</Files>
```

```
<Directory ~ "^\. ">  
    Deny from all  
    AllowOverride None  
</Directory>
```

Apache configuration (httpd.conf)

- Άρνηση πρόσβασης σε φακέλους διαχείρισης του Apache.

```
<Directory /home/httpd/conf>  
    Deny from all  
    AllowOverride None  
</Directory>
```

```
<Directory /home/httpd/logs>  
    Deny from all  
    AllowOverride None  
</Directory>
```

```
<Directory /home/httpd/SSL>  
    Deny from all  
    AllowOverride None  
</Directory>
```

Apache configuration (httpd.conf)

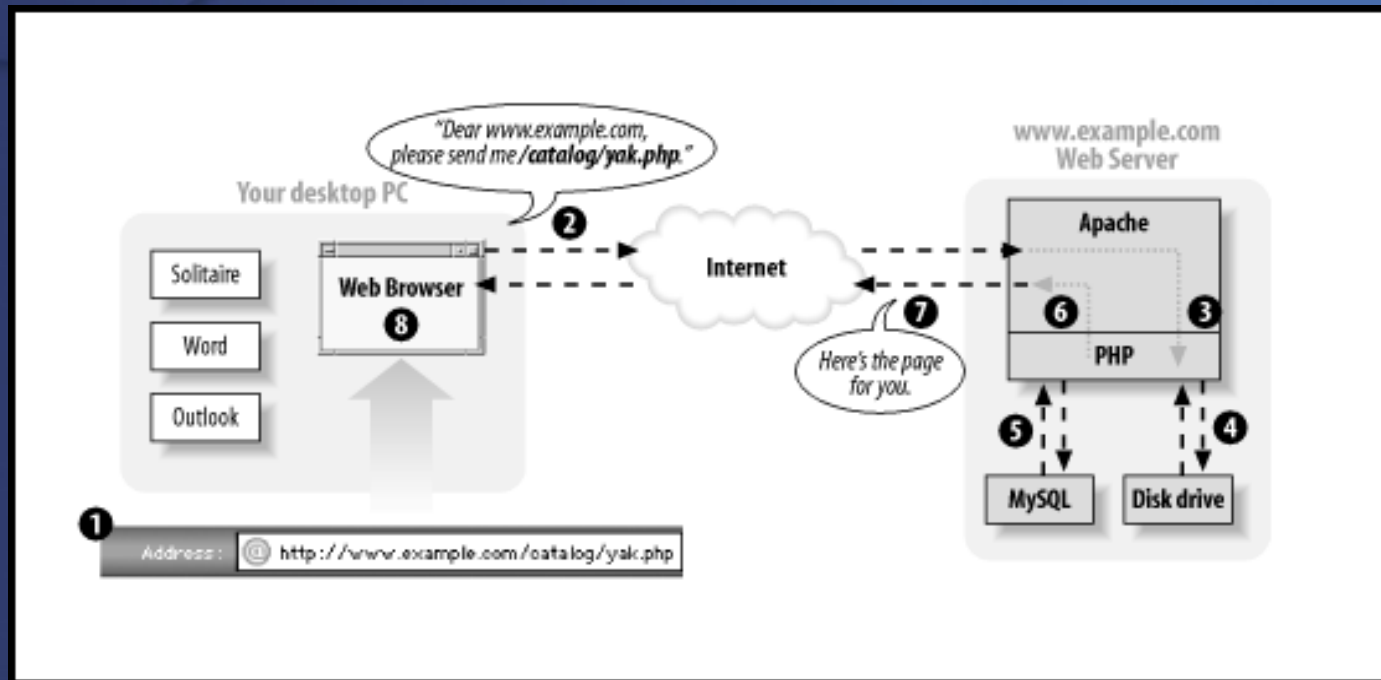
- Άρνηση πρόσβασης σε φακέλους διαχείρισης της PHP.

```
<Directory /home/httpd/php_tmp>  
    Deny from all  
    AllowOverride None  
</Directory>
```

```
<Directory /home/httpd/php_includes>  
    Deny from all  
    AllowOverride None  
</Directory>
```

Γενικά για την PHP

- Η PHP είναι μια **Server Side** γλώσσα προγραμματισμού.
(εικόνα από O'Reilly – Learning PHP 5)



Γενικά για την PHP

- Το `php.ini` είναι το βασικό αρχείο ρυθμίσεων της PHP.
- Απαιτεί μεγάλη προσοχή. Λάθος περιεχόμενο ίσως οδηγήσει σε προβλήματα ασφάλειας.
- Η αρχική του τοποθεσία είναι `/usr/lib/php.ini`.
- Κάθε φορά που ξεκινάει ο Apache μαζί με το module της PHP γίνεται ανάγνωση του `php.ini`. Αν κάνουμε αλλαγές πρέπει να κάνουμε restart και τον Apache.

PHP configuration (php.ini)

- Προτεινόμενες γενικές ρυθμίσεις.

```
memory_limit = 4M
```

```
default_mimetype = "text/html"
```

```
default_charset = "US-ASCII"
```

```
include_path = ".:usr/lib/php:/home/httpd/php_includes"
```

```
doc_root = "/home/httpd/sites"
```

```
file_uploads = On
```

```
upload_tmp_dir = "/home/httpd/php_tmp"
```

```
upload_max_filesize = 2M
```

PHP configuration (php.ini)

- Προτεινόμενες ρυθμίσεις ασφάλειας.

expose_php = Off
register_globals = Off

magic_quotes_gpc = On
magic_quotes_runtime = Off
magic_quotes_sysbase = Off

session.cache_limiter = "nocache"
session.bug_compat_warn = On
session.hash_function = 1
session.hash_bits_per_character = 4
session.use_cookies = On
session.use_only_cookies = On
session.cookie_secure = On

PHP configuration (php.ini)

- Προτεινόμενες ρυθμίσεις ασφάλειας.

```
disable_functions = ini_alter,ini_restore,ini_set,restore_include_path,  
set_include_path,set_magic_quotes_runtime,symlink,exec,passthru,  
proc_open,shell_exec,system,popen,chdir,chroot
```

```
safe_mode = On
```

```
safe_mode_gid = Off
```

```
safe_mode_include_dir = /home/httpd/php_includes/
```

```
safe_mode_exec_dir = /home/httpd/nonexistent/
```

```
safe_mode_allowed_env_vars = PHP_,SERVER_
```

```
safe_mode_protected_env_vars = LD_PRELOAD,LD_LIBRARY_PATH
```

Firing up Apache!

- Όλα έτοιμα για να ξεκινήσει ο Apache (επιτέλους) :-)

```
# httpd -f /home/httpd/conf/httpd.conf
```

- Πήγαν όλα καλά ;

```
# httpd -f /home/httpd/conf/httpd.conf
```

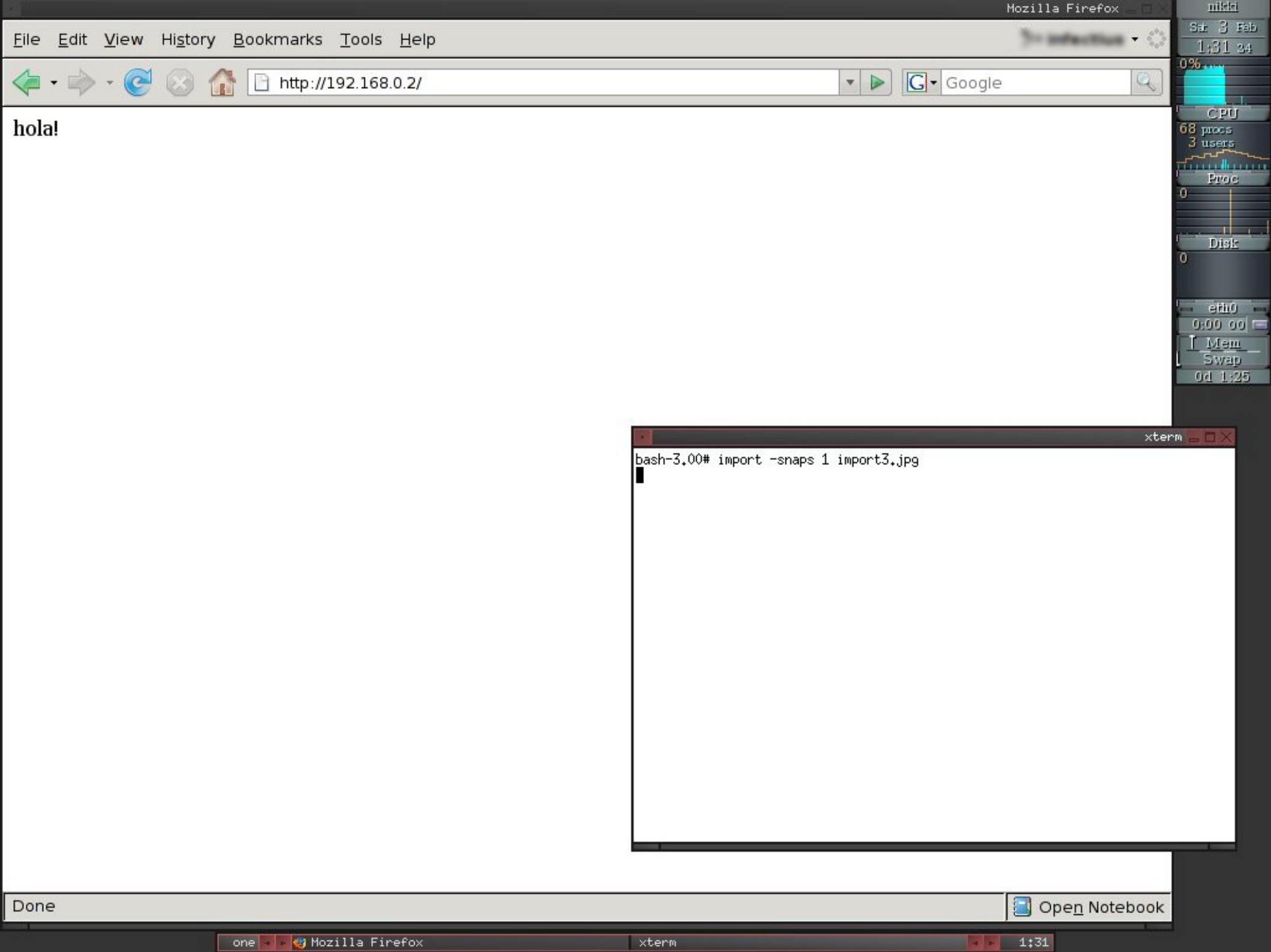
```
# netstat -an | grep LISTEN
```

```
tcp        0      0 0.0.0.0:80          0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:443         0.0.0.0:*          LISTEN
```

```
# su httpd
```

```
$ echo "<html><head></head><body>hola!</body></html>" >  
/home/httpd/sites/localhost/index.html
```

```
$ echo "<html><head></head><body>hola!</body></html>" >  
/home/httpd/sites/localhost_ssl/index.html
```

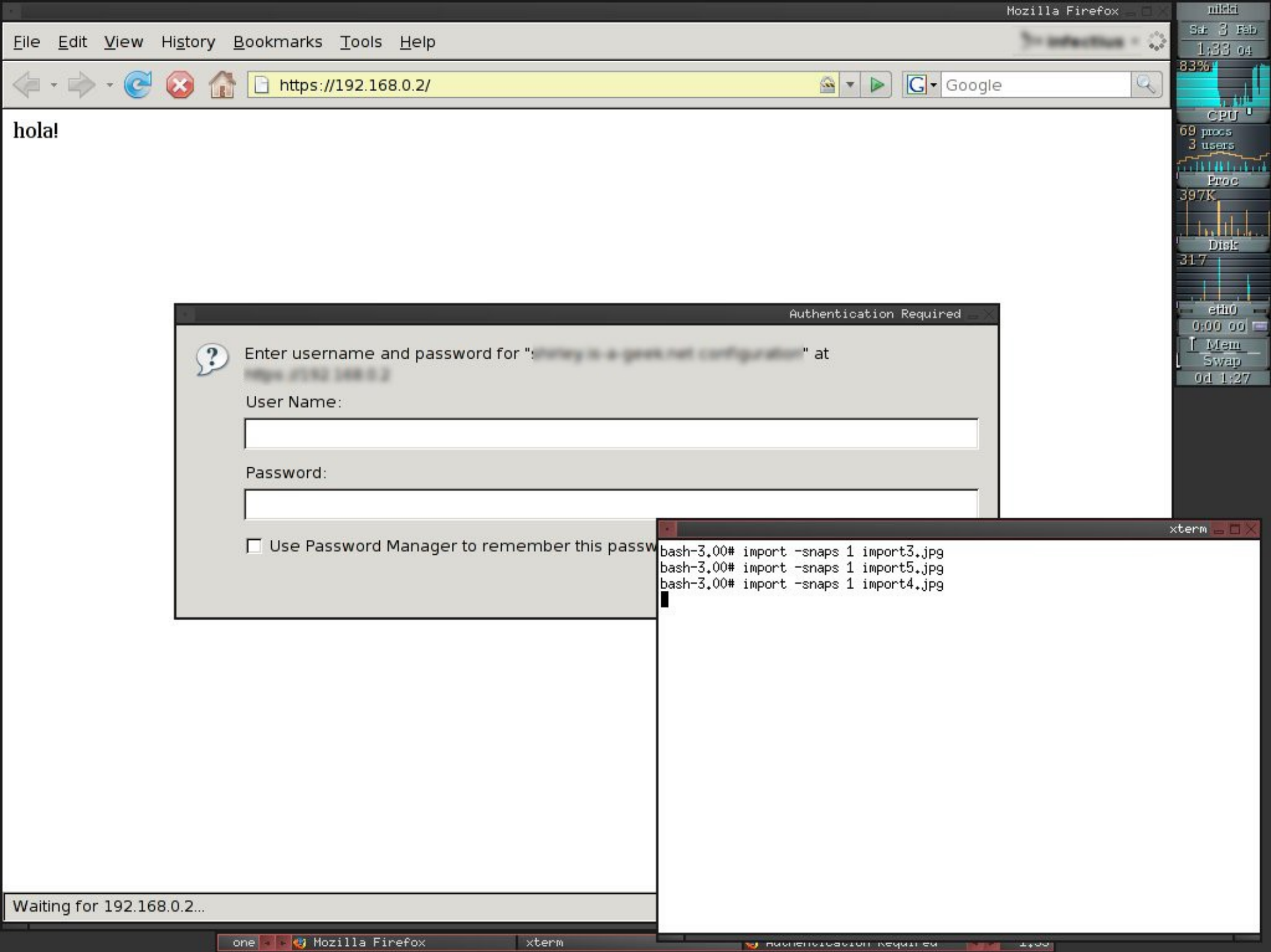


hola!

System monitoring sidebar:

- 0% CPU
- 68 procs, 3 users
- Proc
- 0
- Disk
- 0
- eth0
- 0:00 00
- Mem
- Swap
- 0d 1:25

```
bash-3.00# import -snaps 1 import3.jpg
```



hola!

Authentication Required

Enter username and password for "http://192.168.0.2/" at

User Name:

Password:

☐ Use Password Manager to remember this password

xterm

```
bash-3.00# import -snaps 1 import3.jpg
bash-3.00# import -snaps 1 import5.jpg
bash-3.00# import -snaps 1 import4.jpg
```

SSL MiM

- Μύθος ή πραγματικότητα ;
- Η ιδέα ανοίκει στον stealth του γνωστού group TESO.
- Παρουσιάστηκε για πρώτη φορά στο περιοδικό Phrack #57.
- Ευτυχώς (?) δεν υπάρχει διαθέσιμο εργαλείο ελεύθερο για download.

```

# fetching device eth0 info
# fetched ethernet address 00:02:44:1f:4e:c9
# fetched ip address 10.0.0.11
# starting local area scanning
# skipping our ip address
# scan finished
# no further data to receive
5 hosts detected
[1]: IP: 10.0.0.11 ETHERNET: 00:02:44:1f:4e:c9
[2]: IP: 10.0.0.1 ETHERNET: 08:00:46:d8:99:95
[3]: IP: 10.0.0.12 ETHERNET: 00:0d:88:23:dc:0a
[4]: IP: 10.0.0.33 ETHERNET: 00:0a:e4:a4:16:a6
[5]: IP: 10.0.0.138 ETHERNET: 00:0e:50:2f:b4:d8

Five hosts are currently up and running. The host 10.0.0.11 is our
server address. Flareback sniffing is a done, let's do what we
to present you here the layout of an encrypted irc chat session.

# At this point a fucking error problem that does the network and
# hosts are not up.

root@shirley:/Projects/dns# ./ -dnsarp-vixen -i eth0 -l 10
# fetching device eth0 info
# fetched ethernet address 00:02:44:1f:4e:c9
# fetched ip address 10.0.0.11
# ip redirecting is currently enabled
# enabling ip redirecting
# starting local area scanning
# skipping our ip address
# scan finished
# no further data to receive
3 hosts detected
[1]: IP: 10.0.0.11 ETHERNET: 00:02:44:1f:4e:c9
[2]: IP: 10.0.0.12 ETHERNET: 00:0d:88:23:dc:0a
[3]: IP: 10.0.0.138 ETHERNET: 00:0e:50:2f:b4:d8
enter host to exploit: 2
# 1 threads are going to be created
# thread - sending packets to 10.0.0.138
# final thread handles the target
# 2 threads currently running
[1]: ID: 16386
[2]: ID: 32771
enter host to dns-spoof: 2
# initializing dns spoofing
querying dns server 195.170.2.1:53/udp
sending dns request
receiving dns server response
259 bytes returned
# dns response arrived
# Question entries: 1
# Responses: 1
# Authority records: 5
# Additional records: 5
# Flags: 0x8180

```

```

root@shirley:/Projects/dns# ./ -dnsarp-vixen -i eth0 -l 10
# fetching device eth0 info
# fetched ethernet address 00:02:44:1f:4e:c9
# fetched ip address 10.0.0.11
# starting local area scanning
# skipping our ip address
# scan finished
# no further data to receive
5 hosts detected
[1]: IP: 10.0.0.11 ETHERNET: 00:02:44:1f:4e:c9
[2]: IP: 10.0.0.1 ETHERNET: 08:00:46:d8:99:95
[3]: IP: 10.0.0.12 ETHERNET: 00:0d:88:23:dc:0a
[4]: IP: 10.0.0.33 ETHERNET: 00:0a:e4:a4:16:a6
[5]: IP: 10.0.0.138 ETHERNET: 00:0e:50:2f:b4:d8

Five hosts are currently up and running. The host 10.0.0.11 is our
server address. Flareback sniffing is a done, let's do what we
to present you here the layout of an encrypted irc chat session.

# At this point a fucking error problem that does the network and
# hosts are not up.

root@shirley:/Projects/dns# ./ -dnsarp-vixen -i eth0 -l 10
# fetching device eth0 info
# fetched ethernet address 00:02:44:1f:4e:c9
# fetched ip address 10.0.0.11
# ip redirecting is currently enabled
# enabling ip redirecting
# starting local area scanning
# skipping our ip address
# scan finished
# no further data to receive
3 hosts detected
[1]: IP: 10.0.0.11 ETHERNET: 00:02:44:1f:4e:c9
[2]: IP: 10.0.0.12 ETHERNET: 00:0d:88:23:dc:0a
[3]: IP: 10.0.0.138 ETHERNET: 00:0e:50:2f:b4:d8
enter host to exploit: 2
# 1 threads are going to be created
# thread - sending packets to 10.0.0.138
# final thread handles the target
# 2 threads currently running
[1]: ID: 16386
[2]: ID: 32771
enter host to dns-spoof: 2
# initializing dns spoofing
querying dns server 195.170.2.1:53/udp
sending dns request
receiving dns server response
259 bytes returned
# dns response arrived
# Question entries: 1
# Responses: 1
# Authority records: 5
# Additional records: 5
# Flags: 0x8180

```

nltk
Sat 3 Feb
12:57 19
3%

CPU
59 procs
3 users

Proc
328K

Disk
0

eth0
0

eth2
0:00 00

Mem
Swap
0d 0:51


```

#snap value      number of screen snapshots
bash-3.00# import -snap 1 import1.jpg
import1: unrecognized option '-snap'.
bash-3.00# import -snap 1 import1.jpg
bash-3.00# ls -l
total 800
-rw-rw-r-- 1 root root 200000 2006-07-27 14:08 001.jpg.snap_25
-rw-rw-r-- 1 root root 0 2006-07-27 14:08 0.snap
-rw-rw-r-- 1 root root 96 2006-07-28 17:08 001.jpg.snap
-rw-rw-r-- 1 root root 96 2006-08-01 17:08 001.jpg.snap
-rw-rw-r-- 1 root root 96 2006-08-01 17:08 001.jpg.snap
-rw-rw-r-- 1 root root 200 2006-08-01 17:08 001.jpg.snap
-rw-rw-r-- 1 root root 200000 2007-07-27 09:07 import1.jpg
-rw-rw-r-- 1 root root 200 2006-08-01 17:08 import1.jpg
-rw-rw-r-- 1 root root 200 2006-08-01 17:08 001.jpg
-rw-rw-r-- 1 root root 2000 2006-07-28 17:08 001.jpg.snap
-rw-rw-r-- 1 root root 200 2006-08-01 17:08 001.jpg.snap
-rw-rw-r-- 1 root root 200000 2006-08-01 17:08 001.jpg.snap
-rw-rw-r-- 1 root root 200 2006-08-01 17:08 001.jpg
-rw-rw-r-- 1 root root 96 2006-07-28 17:08 001.jpg
-rw-rw-r-- 1 root root 200 2007-07-27 09:07 001.jpg
-rw-rw-r-- 1 root root 200000 2007-07-27 09:08 001.jpg.snap
bash-3.00# import -snap 1 import2.jpg

```

```
bash-3.00#
```



SSL MiM

[SENT]

GET / HTTP/1.1

Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
application/x-shockwave-flash, application/vnd.ms-powerpoint,
application/vnd.ms-excel, application/msword, */*

Accept-Language: el

Accept-Encoding: gzip, deflate

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1;
SV1; .NET CLR 2.0.50727)

Host: xxx.xxx.xxx.xxx

Connection: Keep-Alive

Authorization: Basic aGsvd2hhdHRoZWZ1Y2s/DQo=