

# Installare o Aggiornare itaEngine LINUX/WINDOWS

Questo wiki raccoglie la descrizione delle diverse attività che concorrono all'installazione di itaEngine. Il wiki è in fase di costruzione quindi allo stato attuale prevalentemente una raccolta di appunti da consultare secondo le esigenze del sistemista. si è cercato comunque di rispettare la sequenza temporale migliore per ottenere una corretta installazione delle procedure.

## Preparazione del server web

### Linux

Si parte dal presupposto che si disponga di una installazione lamp su una distribuzione linux quale:

- Suse >= 11.4
- Debian
- Centos
- Redhat

Altre distribuzioni non sono state utilizzate ma fondamentalmente si necessita dei pacchetti elencati:

Apache Http server 2.2 >=

php 5.x =

Jre 1.6 >=

### Verifica Autorizzazioni SELINUX

Nei casi in cui si attivo il pacchetto SELINUX Disabilitare come segue:

```
vi /etc/sysconfig/selinux
```

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
# enforcing – SELinux security policy is enforced.
# permissive – SELinux prints warnings instead of enforcing.
# disabled – No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
# targeted – Targeted processes are protected,
# mls – Multi Level Security protection.
SELINUXTYPE=targeted
```

Riavviare il sistema oppure:

```
# setenforce 0
```

Controllare se SELinux è stato correttamente disattivato

```
# getenforce
```

## Configurare Apache

- Editare un nuovo file di configurazione di apache es. ***/etc/apache2/conf.d/itaEngine.conf*** e predisporre una cartella web che ospiti l'applicativo itaEngine:

httpd 2.4

```
Alias /itaEngine "E:/Cityware.Online/xampp/htdocs/itaEngine"
<Directory E:/Cityware.Online/xampp/htdocs/itaEngine>
    AllowOverride None
    Options -Indexes +FollowSymLinks +ExecCGI
    Order deny,allow
    Allow from all
</Directory>
```

httpd 2.2

```
Alias /itaEngine/ "d:/Cityware.online/xampp/htdocs/itaEngine/"
<Directory d:/Cityware.online/xampp/htdocs/itaEngine/>
    AllowOverride None
    Options -Indexes +FollowSymLinks +ExecCGI
    Order allow,deny
    Allow from all
</Directory>
```

## Moduli PHP

Verificare l'esistenza dei seguenti moduli:

*CentOS* > *php -m*

- apc
- Bz2
- Chilkat
- curl
- dom
- gd
- ftp
- json
- ldap (**Attenzione non sempre presente**)
- libxml
- mbstring
- mysql

- mysqli
- openssl
- simplexml
- soap
- pgsql
- PDO:pdo\_mysql
- PDO:pdo\_postgres
- xml
- xmlreader
- xmlrpc
- xmlwriter
- [Zend Modules]
- Xdebug

Verificare la presenza del pacchetto pear con il comando:\

```
pear list -a
```

Pacchetti:

- mail
- mail\_mime
- mail\_mimedecode

Se mancante installare con gli strumenti messi a disposizione dalla distribuzione linux:

\* yast \* yum \* apt-get \* Etc...

```
pear list -a
Mail_Mime
Mail_mimeDecode
Net_dime
SOAP
pear install Mail_Mime
pear install Mail_mimeDecode
pear install Net_Dime
pear install SOAP-beta
```

## Parametrizzazione php

- Editare il file: **/etc/php5/apache2/php.ini** e gestire i seguenti parametri
  - session.gc\_maxlifetime = 3600 per aumentare o diminuire il tempo di sessione
  - session.gc\_divisor = 1000 per aumentare o diminuire il numero di chiamate a php prima di effettuare il controllo di validità sessione
  - max\_execution\_time = 7200 per aumentare il tempo massimo di esecuzione delle script
  - memory\_limit = 1024M aumentare il picco di memoria ad almeno 1GB
  - upload\_max\_filesize = 20M
  - post\_max\_size = 50M
  - max\_input\_vars = 3000 per aumentare o diminuire il numero di dati in input da elaborare
- In modalità "debug" attivare il display\_error (ricordarsi di disattivarlo in produzione!)

- display\_error on

## Configurazione repository locale

Clonare il repository pubblico:

```
git clone http://itadeployer@srvgitlab.apra.it/itaengine/itaEngine.git
git checkout [branch] (es. sviluppo, pagoPa)
git pull
```

Utente pubblico per accesso a Git: itadeployer / deploy123

esempio del file .git/config

```
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
[remote "origin"]
    fetch = +refs/heads/*:refs/remotes/origin/*
    url =
http://itadeployer:deploy123@srvgitlab.apra.it/itaengine/itaEngine.git
[branch "master"]
    remote = origin
    merge = refs/heads/master
[credential]
    helper = store
```

## phpMyAdmin

Installare l'ultima versione di phpmyadmin ed eventualmente rendere sicura la cartella con le direttive di htaccess o editando il file **/etc/apache2/conf.d/phpMyAdmin.conf**: (esempio)

```
Alias /qaz321 /usr/share/phpMyAdmin
Alias /mko987 /usr/share/phpMyAdmin

<Directory /usr/share/phpMyAdmin/>
    Order Deny,Allow
    Allow from 188.219.234.118
    Allow from 127.0.0.1
</Directory>
```

## Java Virtual Machine

installare una jdk 1.8.045 o superiore, verificare con il comando:

```
java -version
```

Attenzione utilizzare prodotto oracle e non openjdk per maggiore compatibilità

(Documentare i repository)

## Php Database driver

### Configurazione Sql Server su Linux CentOS

Installare pacchetto: php-mssql-5.3.3-5.el6.x86\_64

Moduli php: mssql pdo\_dblib

### Riferimenti esterni altri driver

Per l'installazione dei driver fare riferimento al seguente link:

<http://wikidev.apra.it/itawiki/doku.php?id=sviluppo:multidb>

## Attivazione moduli php

**mbstring** Il modulo mbstring è necessario per far funzionare l'istruzione mb\_detect\_encoding(). Per CentOS è possibile con il seguente comando:

```
yum install php-mbstring
```

Documentazione di riferimento: [mbstring.installation](#)

### Chilkat 9.5.0

Chilkat è utilizzato per convertire un certificato ppk in certificato openssh. Viene utilizzato ad esempio su pagopa per convertire il certificato dell'sftp di efil. Per ora è richiesto solo se presente il modulo pagopa.

Installazione windows: aggiungere extension=chilkat\_9\_5\_0.dll su php.ini e mettere sotto php/ext la dll presa da <https://www.chilkatsoft.com/php.asp>

Installazione linux: seguire le istruzioni in base al tipo di macchina:

<https://www.chilkatsoft.com/php.asp>

### APC

Linux:

```
pecl install apc
```

Utilizzare le opzioni di default

In alternativa le istruzioni per l'installazione: <https://www.electrictoolbox.com/install-apc-php-linux/>

## Xdebug

Linux: scaricare e compilare il pacchetto wget <https://xdebug.org/files/xdebug-2.4.1.tgz>

```
phpize
./configure
make
make install
```

```
; Enable xdebug extension module
zend_extension=/usr/lib64/php/modules/xdebug.so
xdebug.profiler_append = 0
xdebug.profiler_enable = 1
xdebug.profiler_enable_trigger = 0
xdebug.profiler_output_dir = "/tmp"
xdebug.profiler_output_name = "cachegrind.out.%t-%s"
xdebug.profiler_enable = 0
xdebug.profiler_append=0
xdebug.extended_info=1
xdebug.remote_enable=1
xdebug.remote_handler=dbgp
xdebug.remote_mode=req
xdebug.remote_host=127.0.0.1
xdebug.remote_port=9000
xdebug.idekey=netbeans-xdebug
xdebug.trace_output_dir = "/tmp"
xdebug.show_exception_trace=0
xdebug.show_local_vars=9
xdebug.show_mem_delta=0
xdebug.trace_format=0
```

## Documentazione Composer

<http://wikidev.apra.it/itawiki/doku.php?id=itaengine:composer>

## Soap

Per eseguire l'installazione del modulo soap usare:

```
pear install soap
```

in caso di errore specificare il channel con:

```
pear install channel://pear.php.net/SOAP-0.13.0
```

## Configurazione utente FTP

Creazione utente FTP su macchina albonew.cityware.it

Per poter pubblicare da cityware sul portale dell'albo pretorio è necessario configurare un utente FTP

sullo stesso server in cui è stato installato il webservice chiamato da Cityware. L'utente FTP deve avere uid e gid identici all'utente apache (per una spiegazione approfondita si rimanda al wiki completo [https://wiki.archlinux.org/index.php/Users\\_and\\_groups\\_\(Italiano\)](https://wiki.archlinux.org/index.php/Users_and_groups_(Italiano)) )

Creare la home directory per l'utente FTP:

```
# mkdir /home/ftp/<ente>
```

Esempio:

```
# mkdir /home/ftp/montecchio
```

Verificare numero uid e gid dell'utente apache:

```
# cat /etc/passwd | grep <utente-apache>
```

Esempio:

```
# cat /etc/passwd | grep apache | sed s/:/\ /g | awk '{print "UID: " $3  
"\t\nGID: " $4}'
```

```
UID: 48
```

```
GID: 48
```

Creare l'utente ftp con uid e gid identici all'utente apache:

```
# useradd -u <uid> -g <gid> -d /home/ftp/<ente> ftp<ente>
```

Esempio:

```
# useradd -u 48 -g 48 -d /home/ftp/montecchio ftpmontecchio
```

Dare i giusti permessi alla home directory dell'utente:

```
# chown -R ftp<ente>:ftp<ente> /home/ftp/<ente>
```

Esempio:

```
# chown -R ftpmontecchio:ftpmontecchio /home/ftp/montecchio
```

Impostare una password per l'utente:

```
# passwd ftp<ente>
```

Esempio:

```
# passwd ftpmontecchio
```

Testare la configurazione con winscp, provare a caricare un file e a cancellarlo:

```
Host: albonew.cityware.it
```

```
Porta: 21
```

```
Utente: ftp<ente>
```

```
Password: ****
```

## Configurazione Omnis Server

Da interfaccia web itaEngine aprire il pannello di controllo > Configurazione parametri > Parametri Omnis Studio

Impostare i seguenti parametri

```
Protocollo: http
Url Http Server: [indirizzoHttpServer:porta]
Url Omnis Web Server: [indirizzoOmnisServer:porta]
Omnis CGI Name: /cgi-bin/nph-omniscgi.exe
Libreria di default: CITYWARE
Nome Remote Task: RT_HTTP_REQUEST
Nome Remote Task (interattivo): RT_HTTP_REQUEST_INT
Parametri in formato esadecimale: 0
Gestione abilitazione Omnis Server: 1
```



Esempio

From:

<https://wiki.nuvolaitalsoft.it/> - **wiki**

Permanent link:

<https://wiki.nuvolaitalsoft.it/doku.php?id=installazione:itaengine&rev=1510762570>

Last update: **2018/03/19 10:45**

