



Introduzione a WPKG

WPKG è una soluzione open source (costituita da una parte client e da una parte server) pensata per la distribuzione, aggiornamento e rimozione automatici del software in dotazione ai PC di una rete aziendale. Permette la gestione di profili associabili a diverse macchine, in modo da consentire installazioni differenziate a seconda dell'utilizzatore e del settore a cui egli afferisce.

1. Installazione del server

La parte server di WPKG consiste in una semplice condivisione SMB, dove sono caricati i file necessari per il suo funzionamento. La condivisione deve essere accessibile da tutti i PC che utilizzano il client, perlomeno in sola lettura (si può anche creare un utente allo scopo).

L'organizzazione suggerita di tale condivisione è la seguente:

- nella cartella **installer**, si inseriscono tutti i file necessari all'installazione della parte client e della parte server
- nella cartella **sw**, tutti gli installer dei programmi monitorati da WPKG
- nella cartella **wpkg**, i file della parte server necessari al funzionamento di WPKG (la versione iniziale di questi ultimi si trova nel file **installer/wpkg-1.3.1-bin.zip**)

1.1 Software monitorato

Perché sia possibile la gestione di un software tramite WPKG, il relativo installer (caricato opportunamente nella cartella **sw** di cui sopra) deve prevedere la possibilità di eseguire l'installazione in modalità silente, e a livello globale (cioè non di profilo utente), sul PC.

Per molti software disponibili sul mercato, la relativa documentazione è disponibile (in via non ufficiale) a partire dalla pagina [http://wpkg.org/Category:Silent Installers](http://wpkg.org/Category:Silent%20Installers). In questo sito sono presentati anche gli "script di pacchetto" (suscettibili di correzioni necessarie) di WPKG che consentono il monitoraggio del software.



2. Installazione del client

Come client per WPKG è stata scelta una soluzione basata su Wpkg-GP: questa risulta l'unica funzionante su Windows 7. L'installazione originaria prevede l'esecuzione degli script di aggiornamento all'avvio del PC, ma con alcune modifiche al sistema è possibile programmarli allo spegnimento: quest'ultima soluzione è preferibile, un eventuale aggiornamento del software all'avvio del PC comporterebbe ritardi e quindi disagi al personale, specie se non viene avvertito.

2.1 Installazione per esecuzione all'avvio del PC

Installare il client WPKG, lanciando il file **Wpkg-GP-0.17_x86.exe** (per sistemi a 32 bit) o **Wpkg-GP-0.17_x64.exe** (per sistemi a 64 bit) presente nella cartella **installer** sul server WPKG. Completare la procedura guidata inserendo gli opportuni valori per creare la configurazione (in particolare per l'utente e la password che ha accesso alla condivisione degli installer).

Altri client possono essere correttamente inizializzati copiando, da un'altra postazione, il file **Wpkg-GP.ini** presente nella cartella di installazione del client WPKG (di default è %PROGRAMFILES%\Wpkg-GP).

2.2 Installazione per esecuzione allo spegnimento del PC

Completare l'installazione come sopra, quindi

1. Creare il seguente file (con nome **wpkg_shutdown.cmd**) nella cartella di installazione del client WPKG:

```
@echo off
echo ^+-----^+
echo ^| Aggiornamento delle applicazioni installate in corso           ^|
echo ^| Il PC si riavvia/spegne automaticamente, non forzare lo spegnimento... ^|
echo ^+-----^+
"%PROGRAMFILES%\wpkg-GP\wpkgPipeClient.exe" Execute
```

2. Creare il seguente file (con nome **wpkg_disable_boot.reg**) e utilizzarlo per modificare il registro di sistema:

```
Windows Registry Editor Version 5.00

[-HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\windows
NT\CurrentVersion\winlogon\GPExtensions\{A9B8D792-F454-11DE-BA92-FDCF56D89593}]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\windows
NT\CurrentVersion\winlogon\GPExtensions\AutorunsDisabled]
```



```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\windows
NT\CurrentVersion\winlogon\GPExtensions\AutorunsDisabled\{A9B8D792-F454-11DE-BA92-
FDCF56D89593}]
@="wpkg-GP"
"DllName"=hex(2):43,00,3a,00,5c,00,50,00,72,00,6f,00,67,00,72,00,61,00,6d,00,\
20,00,46,00,69,00,6c,00,65,00,73,00,5c,00,57,00,70,00,6b,00,67,00,2d,00,47,\
00,50,00,5c,00,57,00,70,00,6b,00,67,00,2d,00,47,00,50,00,2e,00,64,00,6c,00,\
6c,00,00,00
"NoBackgroundPolicy"=dword:00000001
"NoUserPolicy"=dword:00000001
"ProcessGroupPolicy"="ProcessGroupPolicy"

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\windows\CurrentVersion\Policies\System]
"MaxGPOScriptWait"=dword:00000708
"HideshutdownScripts"=dword:00000000
```

3. Modificare la seguente Group Policy (si accede eseguendo **gpedit.msc**):

- percorso: **Criteri computer locale -> Impostazioni di Windows -> Script (Avvio/Arresto) -> Arresto** (in Windows 7) / **Chiudi sessione** (in Windows XP)
- nome script: **%PROGRAMFILES%\Wpkg-GP\wpkg_shutdown.cmd**
- parametri script: lasciare vuoto

3. Configurazione del server

La parte server di WPKG si configura mediante l'utilizzo di più file XML, che vengono inseriti in varie posizioni della cartella **wpkg**. Le tipologie principali di file sono:

- file di tipo host, presenti nel file **hosts.xml** e nella cartella **hosts**
- file di tipo profilo, presenti nel file **profiles.xml** e nella cartella **profiles**
- file di tipo pacchetto, presenti nel file **packages.xml** e nella cartella **packages**

Per quanto riguarda la logica della configurazione, tralasciando ulteriori dettagli che saranno esplicitati in seguito:

1. si definiscono i file di tipo pacchetto, contenenti le istruzioni necessarie per eseguire l'installazione, l'aggiornamento e la rimozione del software ad esso correlato
2. si definiscono quindi i profili, che vanno collegati ai pacchetti specificati al punto 1: presumibilmente questi rappresentano le possibili diverse mansioni degli operatori
3. infine, si definiscono gli host e si associano ai diversi profili in base alle mansioni assegnate agli operatori che utilizzano il PC



3.1 File di tipo pacchetto

Si veda per maggiori dettagli: <http://wpkg.org/Packages.xml>

Un file di tipo pacchetto comprende le informazioni necessarie alla gestione del software in modo silente.

Nei vari file, per ogni singolo pacchetto, sono definiti principalmente:

1. intestazione del pacchetto (tag <package>), che include:
 - a. identificativo per WPKG (deve essere unico per tutti i file)
 - b. nome completo
 - c. revisione (utilizzato per stabilire l'effettiva necessità dell'aggiornamento)
 - d. priorità (incide sull'ordine di esecuzione)
 - e. necessità di riavvio del sistema
2. eventuali altri pacchetti (dipendenze) da installare prima (tag <depends>), dopo (tag <chain>) o comunque (tag <include>) assieme: vedi per dettagli http://wpkg.org/Package_dependencies
3. alcune variabili (tag <variable>) che è possibile utilizzare nel resto del file per mantenere congrui altri riferimenti, evitando così l'uso di "valori magici"
4. i controlli da effettuare per valutare la necessità dell'installazione o aggiornamento (tag <check>)
5. i comandi da lanciare se il software deve essere installato (tag <install>)
6. i comandi da lanciare se il software deve essere aggiornato (tag <upgrade>)
7. i comandi da lanciare se il software deve essere rimosso (tag <remove>)

3.2 File di tipo profilo

Si veda per maggiori dettagli: <http://wpkg.org/Profiles.xml>

Un file di tipo profilo consente di specificare il software (organizzato nei pacchetti di cui sopra) che va installato e mantenuto aggiornato per una relativa mansione.

La sintassi del file prevede:

1. un'intestazione generale (tag <profiles>), presente comunque su tutti i file di profilo
2. per ogni profilo, un'intestazione comprendente il suo identificativo (tag <profile id="...">): l'identificativo non deve essere ripetuto, nemmeno in file diversi
3. all'interno di ciascun profilo vanno specificati i software/pacchetti ad esso associati (tag <package package-id="..." />). Si può anche specificare una dipendenza da un altro profilo (tag <depends profile-id="..." />), in modo da includere automaticamente tutti i pacchetti inseriti in quest'ultimo



3.3 File di tipo host

Si veda per maggiori dettagli: <http://wpkg.org/Hosts.xml>

Un file di tipo host realizza l'associazione tra i computer e il software che deve essere installato, utilizzando i profili come collegamento.

La sintassi del file prevede un'intestazione generale (tag <hosts:wpkg>), seguita dall'elenco degli host (tag <host name="...">) e dai profili ad essi associati (attributo profile-id="..." se unico, tag <profile profile-id="..."> se multiplo).

Come nome di host è possibile specificare un nome statico di dominio (ottenibile dalla variabile d'ambiente %COMPUTERNAME% sul PC), oppure un'espressione regolare con la quale sia possibile identificare più valori. La sintassi per specificare l'espressione regolare è simile a quella utilizzata da linguaggi come PHP e PERL, le cui regole sono ampiamente documentate in questa pagina: <http://php.net/manual/en/reference.pcre.pattern.syntax.php>.

4. Funzionamento di WPKG

Quando WPKG è in esecuzione, consulta il file **C:\Windows\System32\wpkg.xml** dove sono riportati gli esiti dei controlli del monitoraggio, e i pacchetti software di cui è prevista l'installazione con relativo numero di versione.

Se questo file non esiste, viene creato da WPKG assumendo inizialmente che nessuno dei software monitorati risulta installato sul PC.

4.1 Controlli pre-installazione

Al momento dell'esecuzione:

1. è verificato il nome del computer, quali profili sono associati ad esso e quindi quali pacchetti devono essere presenti sul PC
2. se un pacchetto previsto dalla configurazione server non risulta installato sul PC, vengono effettuati i controlli previsti nelle istruzioni <check>: qualora anche uno solo di questi desse esito negativo, si procede con l'installazione, come indicato nelle istruzioni <install>. In assenza di controlli, si assume che il pacchetto debba comunque essere installato
3. se un pacchetto previsto dalla configurazione server risulta installato sul PC, viene confrontato il numero di versione locale con quello del server (attributo revision): se quest'ultimo risulta successivo, si procede con l'aggiornamento così come indicato nelle istruzioni <upgrade>



giunta regionale

4. se un pacchetto non è previsto nella configurazione server, ed è installato sul PC (secondo la configurazione locale), sarà avviata la procedura di rimozione così come specificata nelle istruzioni <remove>

L'ordine di installazione dei pacchetti, oltre che da eventuali direttive <depends>, <chain> o <include>, è determinato dal valore `priority` presente nell'intestazione del pacchetto: un valore più alto comporta precedenza nell'installazione.

4.2 Controlli post-installazione

Terminato ciascun comando, WPKG esegue con un controllo sul suo esito basandosi sul codice di errore ritornato: se il risultato è 0, l'elaborazione prosegue con il comando successivo, in caso contrario l'elaborazione viene interrotta e si assumerà che il pacchetto non sia stato installato/aggiornato/rimosso. In alcuni casi può essere atteso un codice di errore diverso da zero, ad esempio (per la lista completa si veda <https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381%28v=vs.85%29.aspx>):

- 1 – funzione non valida
- 2 – file non trovato
- 128 – processo non trovato
- 1605 – prodotto sconosciuto (software non installato)
- 3010 – richiesto riavvio del PC

In questo caso, i relativi codici diversi da zero devono essere specificati nella configurazione attraverso un tag <exit code="..."> associato al rispettivo comando.

4.3 Verifica del funzionamento

L'esito di alcuni di questi controlli può essere verificato attraverso il **Visualizzatore Eventi** di Windows: all'interno di **Registri di Windows** -> **Applicazione** risulteranno presenti alcuni record con origine **WSH** riportanti l'esito dei controlli pre- e post-installazione di WPKG (qualora fallissero, si può verificare la presenza di record contigui con origine **MsiInstaller** o comunque relativi allo specifico pacchetto installato).