

Pregaziti konfiguraciju ili ne?

Željko Boroš | 30 kolovoza, 2012

Nadogradnje Debiana se pojavljuju redovito (dobro, uglavnom redovito) svake dvije godine, dakle sistemcima nadogradnja nije nepoznat teren. Ipak, pomisao na nadogradnju poslužitelja mnogima izaziva glavobolju i mučninu, a ponegdje i druge simptome. Kao da čujemo da govore "ajoj, sada opet pola stvari treba ispočetka konfigurirati", "samo da mi web proradi što prije", "valjda radius neće pasti" i tako dalje. Velika većina sistemaca je došla iz svijeta Windows operativnih sustava, gdje su nadogradnje obično manje bolne, prvenstveno zbog *registryja*, raznih *wizarda* i općenito činjenice da većina softvera dolazi iz iste kuće. Na Linuxu to nije tako.

Na Linuxu ne postoji nikakav *registry* (postoje projekti koji takvo što nastoje postići, poput projekta **Elektra**), a svi programi svoje postavke čuvaju u tekstualnim datotekama uglavnom unutar direktorija `/etc`. Sam format datoteke se također razlikuje, neki programi rabe format poput "INI" datoteka, neki rabe "Apache" format, a većina rabi jednostavni "varijabla=vrijednost" format. Sve to samo po sebi, zapravo, i nije problem.

Problem je što ne postoji način nadogradnje iz prethodne inačice konfiguracijske datoteke u drugu. U jednom trenutku neka varijabla postaje "*obsolete*", dakle program je više neće rabiti. Način na koji će se to napraviti je također različit od programa do programa, od inačice do inačice. Jedan će program takve "nevrijedeće" varijable ignorirati (uz upozorenje ili bez njega!), a drugi će jednostavno stati (najnoviji primjer je MySQL 5.1 koji će se odbiti pokrenuti nakon učitavanja varijabli `skip-bdb`, `skip-innodb`, ali i drugih).

Da postoji standardizirani način pisanja konfiguracijskih datoteka, bilo bi moguće napraviti program-parser koji će učitati sve varijable, unijeti sve eventualne promjene i zapisati promjene, uvažavajući promjene koje je korisnik sam napravio. No, ovo je jednostavno nemoguće napraviti za desetke tisuća programa i njihovih programera, te različitih načina zapisa konfiguracijskih datoteka.

Ostaje, dakle, samo ručni način praćenja i izvršavanja promjena, ali tu ipak imamo neku pomoć. Prije svega je tu sam Debianov paketni sustav, koji neće jednostavno pregaziti vaše izmjene u staroj konfiguraciji, nego će pitati što da učini s konfiguracijskom datotekom (stavi novu ili ostavi staru). Nadalje je tu `debconf` koji vam pomoću par pitanja u lijepim menijima pokušava olakšati konfiguraciju pojedinog paketa. Na kraju, tu je sustav CARNetovih paketa, koji konfigurira pakete na način da pokušava postaviti vrijednosti na one s kojima će program proraditi "od prve". To čini postavljajući varijable na vrijednosti koje će raditi unutar CARNet mreže (primjerice, DNSBL poslužitelj), ili je uobičajena vrijednost neodgovarajuća (primjerice postavke memorije u PHP-u).

Moramo spomenuti da postoje i drugi sustavi za konfiguraciju, obično cijele mreže računala, poput programa `cfengine` ili `puppet`. Oni nam ne pomažu u nadogradnji, nego distribuiraju ručno podešene konfiguracijske datoteke na klijente i nisu predmet ovog članka. Ovdje će se baviti samo prvim korakom, osnovnim pitanjem APT-a: Želite li zadržati staru ili instalirati novu konfiguracijsku datoteku? Svima je poznato to pitanje:

Ukoliko odaberemo odgovor "N", dakle želimo zadržati staru konfiguraciju, sačuvat ćemo sve promjene koje smo ranije unijeli u konfiguracijsku datoteku, ali riskiramo da servis ne radi dobro, ukoliko se uopće pokrene.

Ukoliko odaberemo odgovor "Y", "pregazit" ćemo staru datoteku novom, onom iz paketa. Ovo znači da su vaše promjene izgubljene, a servis će biti pokrenut s inicijalnim vrijednostima.

Postoje još dvije opcije, jedna je ("D") da se pomoću programa `diff` pokažu promjene između dvije datoteke, a druga nudi pokretanje shella - korisničke ljuške ("Z") kako bi mogli vidjeti što je najbolje učiniti u konkretnoj situaciji.

Program `diff` smo vam opisali u članku [/sys-trek-objave/naredbe-za-koje-mozda-niste-znali/#naredbe-](#)

[za-koje-mozda-niste-znali-10-diff-diff3-comm-cmp](#), a on omogućuje prikaz razlika između dvije datoteke. Dakle, u našem slučaju, moći ćete vidjeti što je različito u dvije konfiguracijske datoteke. To je moguće zato jer se konfiguracijska datoteka zapravo nikada ne briše. Ukoliko želite zadržati staru konfiguracijsku datoteku, nova će biti snimljena odmah pored nje, samo s nastavkom .dpkg-dist:

Ukoliko ste odabrali novu konfiguracijsku datoteku, stara će biti preimenovana, odnosno samo će dobiti nastavak .dpkg-old:

Dakle, vaše promjene teško mogu biti izgubljene, a zbog ovog mehanizma, moguće su ostale dvije opcije u APT izborniku (diff i shell).

No, sve ovo ne govori nam koje opcije treba izbrisati, koje dodati, a koje izmijeniti tako da odgovaraju našim potrebama, a istovremeno su aktualne za trenutnu inačicu programa.

Ovdje prestaje svaka pomoć koju vam sustav može pružiti, i dalje je sve na vama. Cijeli se postupak zapravo svodi na iskustvo sistem-inženjera, koji bi trebao:

a) znati sve modifikacije koje je napravio.

Ovo je moguće ukoliko se vodi dnevnik o sustavu, dobro je i u papirnatom obliku.

b) biti dobro upoznat s "bitnim opcijama bitnih servisa". Naravno da je nemoguće zapamtiti sve opcije, ali gdje se podešava veličina e-mail poruka u Postfixu bi trebao znati svatko.

c) moći prepoznati jesu li promjene u novoj konfiguracijskoj datoteci bitne za funkcioniranje servisa ili ne. U principu je poželjno, barem kod ovakvih servisa i programa, staviti novu inačicu konfiguracije. Često su jedine promjene komentari ili pojašnjenja varijabli, uz zgodan dodatak da će kod sljedećih nadogradnji ispis diffa biti manji nego u odnosu na inačice prije aktualne.

d) znati koje promjene donose CARNetovi paketi. Neki paketi bez pitanja prepisuju neke opcije unutar konfiguracije (primjerice, postfix-cn bez ikakvih pitanja prepisuje vrijednosti myhostname, mydomain, myorigin...) kako bi omogućili početno funkcioniranje servisa bez obzira što sistem-inženjer napravio. Kasnije je lako popraviti ono što ne valja.

Da zaključimo... nema generalnog pravila koju opciju odabrati, jer sve ovisi o pojedinom paketu, iskustvu onoga tko radi nadogradnju i raspoloživom vremenu. Čovjek s više iskustva će vjerojatno probleme rješavati odmah, pa će nakon reboota stroj biti odmah spreman. Drugima će biti lakše da proradi barem dio servisa (i skinu korisnike "s vrata"), pa se posvete onima koji još ne rade kako treba.

Kod dugačkih konfiguracijskih datoteka laganu prednost ima opcija ostavljanja stare datoteke, pa upisivanje promjena iz nove datoteke. U slučaju kratkih konfiguracijskih datoteka, vjerojatno je lakše staviti odmah novu datoteku, pa prebaciti tih par opcija iz stare konfiguracijske datoteke.

Jedino što je sigurno je da poznavanje konfiguracije vama važnih servisa, praćenje svih logova i proučavanje dokumentacije i zapisivanje važnih činjenica i parametara skraćuje vrijeme nadogradnji i razine stresa.