

Posudek oponenta k diplomové práci

předložené na Matematicko-fyzikální fakultě
Univerzity Karlovy v Praze

Autor/ka: Petr Šťastný

Název práce: Monitoring DNS serverů domén druhé úrovně

Studijní program a obor: Informatika, softwarové systémy

Rok odevzdání: 2010

Jméno oponenta: Dan Lukeš

Pracoviště: SISAL MFF UK

Práce navazuje na bakalářskou práci téhož autora a ten využívá i zkušenosti z provozu služby, která testování doménových záznamů a provozovaných služeb veřejně nabízí. To je skvělý startovní bod a příslib značných zkušeností autora s oborem práce. Bohužel, patrně právě toto zázemí je v pozadí i stinné stránky této práce, a tou je jistá míra „provozní slepoty“. Veřejně provozovaná služba pracuje tak, že pro konkrétně zadané jméno domény provede ucelenou sadu testů a vydá výsledek. Tato práce se zabývá prováděním obdobných testů pro velmi velké množství domén.

K tomuto účelu autor navrhl a implementoval framework pro správu a distribuci úloh. Ten patří k nejcennější části práce a je navržen natolik univerzálně, že potřeby práce přesahuje. I implementaci lze považovat za dostatečně kvalitní, kód je přehledný a povětšinou dobře pochopitelný. Autor se zabýval a uspokojivým způsobem vyřešil většinu problémů, který s paralelním zpracováním souvisejí včetně problému „vypnutých/ztracených uzlů“.

Podstatně vážnější připomínkou je nevyhovující dokumentace. Autor v práci má popsané základní principy fungování a vedle toho má dokumentaci strojově generovanou z komentářů ve zdrojových kódech. Generovaná dokumentace sice nabízí perfektní a detailní popis každé konkrétní komponenty, obě dokumentace však na sebe prakticky nenasazují. Ten, kdo by chtěl modifikovat některou část programu či doplnit nějakou funkci se sice z principů fungování dozví jak složitá může požadovaná modifikace být, ani jedna dokumentace mu ale příliš nepomůže najít ty části kódu, které bude nutné upravit. Na to je popis v kapitolách 5.x příliš obecný a strojově generovaná dokumentace neobsahuje informace o vzájemných vazbách vůbec.

Vážnou formální vadou práce může být to, že podle údajů uvedených přímo v některých souborech není autor práce autorem celého odevzdaného kódu. V řadě souborů (nejméně v adresáři `phpf/class/gui`) nejenže není uveden jako autor kódu b daném souboru, ale není uveden ani jako ten, kdo soubor naposledy editoval v repository. Přitom jsem ani v dokumentaci ani v textu práce nenašel zmínku o tom, že odevzdaný kód je prací kolektivní. Ani v seznamu zdrojů není externí zdroj nebo spoluautor zmíněn. A soubory nejsou ani vyčleněny do samostatného adresáře „`phpf/external`“, který se zdá být (tiše) dedikován právě cizím komponentám.

Vrátím-li se zpět k implementaci, musím se vrátit i k tomu, že po této stránce je framework zdařilý a kvalitní. Musím se ale také vrátit k zmíněné „provozní slepotě“ autora, který svůj vlastní kvalitní framework následně nedokázal optimálně použít.

Jak plyne z prezentovaných výsledků, víc cca třetina domén v zóně .cz je hostována na méně než deseti DNS serverech a v TLD .sk je situace podobná. Není důvod se domnívat, že by v jiných TLD měla být situace významě odlišná. Autor však nedokázal překročit stín WWW aplikace, kterou provozuje a i v tomto případě k problému přistupuje po jednotlivých doménách. Přestože v několika desítkách tisíc případů bude testovat stále tytéž servery, přistupuje k těmto testům jako k izolovaným případům místo toho aby nejprve zjistil seznam serverů a ohledem na velké množství duplicit naplánoval optimální sadu testů. Ano - řada testů, byť stejného serveru, závisí na tom jaká doména je testována, řada ale nikoliv. Celkovou dobu testování přitom významě prodlužují zejména servery, které se nedaří kontaktovat vůbec – a zde je podstatným rozdílem, jestli se o kontaktování pokouším mnohatisíckrát, nebo jen jednou s platností pro všechny budoucí případy. Prakticky totéž by bylo možné říct o testu na existenci reverzního záznamu pro IP daného server nebo o dopředném resolení pravých stran MX záznamu.

Nejde jen o to, že opakované testování stejných věcí není efektivní – jde i o to, že servery mohou masivní testování vyhodnotit jako útok, zablokovat přístup a tím znehodnotit výsledky následných testů. Autor si sice tento problém uvědomil, ale neprovedl žádnou úvahu o tom, jak reagovat - a odevzdaná implementace vytváří dobré podmínky pro to, aby se zmíněný problém mohl projevit v síle maximální. Autor se navíc ani nepokouší omezit množství testů, které klienti směřují na jeden konkrétní cílový server v jeden okamžik a tak by v nepříznivém případě mohlo testování způsobit testovaným serverům reálné potíže.

Mrzuté je, že autor z časových důvodů velmi omezil sadu testů. Autor testy neprovedl ani nějakém menším vzorku S těmi výsledky, které práce v současné podobě prezentuje tak nelze rozumně posoudit, zda je test v té podobě v jaké byl implementován „jen“ zbytečně pomalý, nebo jestli v praxi není reálně nepoužitelný. Jsem přesvědčen, že jedním z důvodů nepříjemně dlouhé doby trvání testů WWW a SMTP serverů je právě nevhodná paralelizace, ale tím, že autor nic takového ani v omezené míře nezkusil zůstává tento názor na úrovni doměnky.

Je ale třeba zdůraznit, že problém „duplicitních testů“ není vadou návrhu paralelizačního frameworku, kde setrvávám na názoru, že jeho návrh i implementace je kvalitní. Jde o problém následného použití frameworku, konkrétně rozvržení komplexního testu na paralelizované podúlohy. I z hlediska objemu práce mám za stěžejní právě tento framework. A ten mě také přesvědčil, že přes všechny výhrady by práce měla být obhájena. Nicméně, suboptimální užití frameworku a tomu, že autor neprovedl testy, které by (i jemu) ukázaly zda je výsledný produkt vůbec použitelný na testy pro jejichž provádění měl být systém navržen považuji za odpovídající známku „3“.

Datum: 27. ledna 2011

Podpis:



Dan Lukeš