

**Testen van de cloud? Vergeet OTAP! We ontkomen er niet meer aan om te testen in de real life omgeving. Dat kan natuurlijk nu al, voor zover het kleinschalige overzichtelijke transacties betreft: even een bloemetje bestellen en als hij vanmiddag inderdaad wordt afgeleverd heb je de hele 'Order to Cash' keten doorgefloten.**

# Hoe maken we de cloud testbaar?

## Testverkeer veilig de cloud in sturen

**M**aar met ingewikkelde vertakte processen en bijbehorende complexe testscenario's wordt het een ander verhaal. Dat lukt alleen echt goed als ketencomponenten 'Designed for Test' zijn. Ook dat is niets nieuws; hardware (elektronica), auto's, vliegtuigen en wapensystemen worden vanaf het begin ontworpen met allerlei voorzieningen voor veilig testen van 'the real thing'.

### Veilig testen ...

In de militaire wereld kan vrijwel de hele commando- en vuurleidingsketen in de echte omgeving worden ge-dry-run'd. Er zijn alleen uitermate goede en veilige voorzorgen getroffen om te voorkomen dat helemaal aan het einde van de keten die raket echt de lucht in gaat.

Idem dito in de bancaire wereld: in de productieomgeving vinden we testtransacties, testrekeningen, testklanten, enzovoort. Auditors en controllers schrikken daar helemaal niet van, ze zijn het al dertig jaar gewend. Dat gaat allemaal goed omdat de systemen en de protocollen vanaf het begin zijn ontworpen om dit mogelijk te maken. Een testtransactie zal nooit een daadwerkelijke financiële stroom opleveren en kan de rapportages (voor De Nederlandsche Bank bijvoorbeeld) niet vervuilen. Bovendien zijn er voldoende waarborgen tegen fraude. Er zijn geen 'vulnerabilities', om een term uit de webwereld te gebruiken.

### ... in productie

Voor ketencomponenten en services in de cloud geldt dat alles in veel mindere mate. En dat moet veranderen! Hoe kunnen we het zo bouwen dat we

testberichten door de keten kunnen sturen zonder risico's op verstoringen en vervuiling van productiedata? Hoe kunnen we bijdragen aan cloud services en webservices waarmee we veilig kunnen testen in productie?

Ik stel een aantal maatregelen voor waarmee we op een veilige manier testverkeer de cloud in kunnen sturen. Met de zekerheid dat ook in het geval van onverwachte routing, onbedoelde operaties en onvoorziene effecten geen rampen gebeuren. Want dat is natuurlijk wel de essentie van testen: er gebeuren onverwachte dingen. Even tussendoor: om precies dezelfde reden is exploratief testen zo'n waardevolle aanvulling: 50% van de 'bugs found' (OK, 'bevindingen', voor de puristen onder ons) had je van tevoren nooit kunnen bedenken. Maar dit terzijde. Er gebeuren dus onverwachte dingen en het zou mooi zijn als in de cloud voldoende intelligentie ingebouwd is om de betreffende transacties of datastromen naar een 'garbage collector' of iets dergelijks af te buigen. In de veilige wetenschap: geen man overboord en geen proces verstoord, want het zijn slechts testdata. De tester hoeft dan niet overmatig terughoudend en voorzichtig te zijn: hij of zij kan rustig de uithoeken en excepties opzoeken in de veilige wetenschap dat de cloud robuust genoeg is. Maar dan moet het testverkeer wel als zodanig herkenbaar zijn.

### 'Test taxonomie'

Een eigen XML schema, een eigen XBRL 'test taxonomie' lijkt me helemaal niet zo gek. Met extra voorzieningen voor geavanceerd testverkeer, metatags, trace opties en uiteraard met adequate beveili

*Lees verder op pagina 24*

»



**Egbert Bouman**

Valori

egbertbouman@valori.nl

**De aandacht die de cloud krijgt, heeft alle kenmerken van een hype, maar dat neemt niet weg dat het een wezenlijke ontwikkeling is van de manier waarop ICT-diensten worden verleend. Het afnemen van een cloud service is steeds vaker aantrekkelijker dan eigen software maken of kopen.**

# De cloud in testperspectief

## ...en testen in cloudperspectief

In de cloud wordt er alleen betaald voor wat er wordt gebruikt (pay-per-use) en er kan flexibel capaciteit worden bij- of afgeschakeld (elasticiteit). Kortom, steeds meer ICT verdwijnt in de cloud en dat heeft invloed op testen en de rol van de testmanager. Cloud computing introduceert nieuwe eisen en risico's waarvoor een speciale testaanpak nodig is. Hoe test je bijvoorbeeld elasticiteit?.

Testen zelf verandert ook! Zo is er een significante rol weggelegd voor de testmanager in het selectieproces. Dat is een belangrijke fase om tijdig vast te stellen wat voor risico's de gebruiker of de business loopt bij de keuze voor een bepaalde cloud service. Die risico's kunnen liggen op allerlei vlakken. Dat is ook een verandering bij het testen: een accentverschuiving naar niet-functionele aspecten. Het zoeken naar functionele fouten in de software verschuift in de cloud naar de achtergrond.

### Nieuwe eisen en risico's

Cloud computing is volop in ontwikkeling. Het is dan ook een illusie om te verwachten dat voor alle nieuwe eisen en risico's al pasklare testoplossingen beschikbaar zijn. Dit artikel beschrijft een aantal voorbeelden van cloud-specifieke risico's en bijbehorende testaspecten en is een beknopt fundament voor verdere ontwikkeling.

Cloud computing omvat nogal wat. Kijk voor een goed bruikbare definitie op de website van het Amerikaanse NIST ([www.nist.gov/itl/cloud/index.cfm](http://www.nist.gov/itl/cloud/index.cfm)).

Krachtige features van cloud computing zijn elasticiteit en pay-per-use. Is er meer capaciteit nodig, bijvoorbeeld omdat er meer gebruikers zijn, dan kan dat on-demand worden geleverd. De gebruiker ziet

dit vanzelf terug in de afrekening. Maar hoe kan er getest worden of dit in de praktijk ook werkt? In de aanpak hiervoor is een combinatie nodig van load testen (wisselend gebruik nabootsen en zien dat er capaciteit bij komt en weer af gaat), grenswaardenanalyse (wat gebeurt er op de grens van mijn 'bundel') en procescyclustest (het handmatige aanvraagproces en de financiële afrekening).

### Risico's

Met de transitie naar de cloud storten alle risico's van het internet zich over de afnemer heen. Wat voorheen geen issue leek (de server stond in het eigen gebouw), wordt in één keer een punt van aandacht. Wie kan er allemaal bij mijn gegevens in de cloud? Hoe veilig is het transport eigenlijk? Dit leidt niet tot nieuwe soorten beveiligingstesten, maar security eist wel een nog belangrijkere plek op bij het testen.

In de public cloud kan de service trager worden als gevolg van de (piek)belasting van andere afnemers. Als de leverancier een abonneemodel hanteert waarbij de service wordt overboekt (dat levert meer winst op), dan is dat een goede reden om een real-time loadtest uit te voeren. Daarmee wordt bedoeld: het uitvoeren van de loadtest op het moment van de dag dat de piekbelasting zich naar verwachting zal voordoen.

### Acceptatiecriteria

Cloud services worden mondiaal aangeboden en daarmee komt een nieuw pakket van acceptatiecriteria om de hoek kijken: voldoet de cloud service aan allerlei wet- en regelgeving? Zo is onder meer

*Lees verder op pagina 24* »



**Jeroen Mengerink**  
Polteq Test Services  
[jeroen.mengerink@polteq.com](mailto:jeroen.mengerink@polteq.com)

# De cloud testbaar

- » ging zonder introductie van kwetsbaarheden. Misschien is een eigen taal voor testdoeleinden wel een optie: 'XCTL, Extensible Cloud Test Language', voor het veilig uitwisselen van testtransacties. Klinkt goed toch? Daarvoor is wel het een en ander nodig. Een convenant tussen overheid en cloud partijen voor het uitwisselen van testverkeer, met inachtneming van wet- en regelgeving. Misschien moeten de verenigingen van IT-auditors (ISACA, ITGI) ook aanhaken.

## Een volgende stap

Ik geef het toe: het is een wild en pril idee en mijn redenering is natuurlijk veel te simpel. Of juist niet: misschien is het er allemaal al lang en heb ik niet lang genoeg gezocht en met te weinig specialisten gesproken. Maar op de 'IT-verjaardagen' waarop ik dit tussen borrel en bitterballen door heb geopperd is het idee nog niet werkelijk weggehoond, integendeel. Ook tijdens een presenta-

tie voor de ASL/BiSL community (ik geef toe: niet de echte wolkenridders) heb ik de rotte eieren en tomaten gemist.

Dus misschien toch tijd voor een volgende stap, waarbij de test community een drijvende kracht kan zijn. Tien, vijftien jaar geleden lag dat veel moeilijker. Toen was de testwereld nog een naar binnen gerichte club van gelijkhebbers. Als ik me niet sterk vergis, is dat behoorlijk veranderd: testers zijn partner geworden. Mede dankzij de agile trend zijn we, al dan niet tegen wil en dank, uit onze ivoren toren in de duivelsdriehoek gestapt. En maken we ons samen met opdrachtgevers, gebruikers en techneuten druk om tijd, geld en – jawel – kwaliteit.

Vanuit deze positie wordt het tijd om krachtig positie te nemen in de cloud. Wil je kwaliteit, wil je garanties, wil je de complexiteit en de risico's ook in de toekomst beheersen? Dan vinden wij testers dat de cloud verrijkt moet worden met voorzieningen voor veilig testen in productie! «

# Testperspectief

- » het bewaren van privacygevoelige gegevens aan allerlei wetten en regels gebonden. De testmanager zorgt ervoor dat er, in samenspraak met bijvoorbeeld een jurist, een checklist van wettelijke controlepunten wordt samengesteld. Bij het afvinken daarvan kunnen er complicaties aan het licht komen, zoals conflicterende of onduidelijke wetgeving in bepaalde landen en de (on)betrouwbaarheid van de overheid waar de cloud provider is gevestigd. Dit kan tot verrassingen leiden, zoals in het voorbeeld van de Amerikaanse overheid die de Patriot Act in stelling bracht om toegang te krijgen tot de gegevens van WikiLeaks.

## Beschikbaarheid

Wanneer een belangrijk business proces afhangt van een cloud service, is de beschikbaarheid van de service van groot belang. Als een cloud service leverancier (onaangekondigd) een wijziging aanbrengt in de service, kan dat verstrekende gevolgen hebben, zoals een storing in de koppeling met andere systemen of handleidingen die plotseling niet meer kloppen. In een end-to-end testopstelling kan de impact van wijzigingen in een cloud service, maar ook in aangesloten systemen, worden getest. Er is een groeiende noodzaak voor een continu draaiende, end-to-end regressietest. Deze test moet niet alleen op functioneel gebied inzicht in de risico's bieden, maar ook op non-functionele aspecten, zoals performance. De taak van de testmanager stopt dus niet na de implementatie, want ook in de operatie moet er continu worden getest.

ten, zoals performance. De taak van de testmanager stopt dus niet na de implementatie, want ook in de operatie moet er continu worden getest.

## Leveranciersaspecten

De afnemer van cloud services krijgt te maken met allerlei leveranciersaspecten. Wat gebeurt er bijvoorbeeld precies met de gegevens in de cloud als de leverancier failliet gaat? Is er een back-up scenario en, zo ja, is dat getest? Ook de migratie naar de cloud en van de ene cloud service (leverancier) naar een andere moet worden getest. Als er geen geteste migratiestrategie is, levert een afnemer zich over aan één leverancier. Fenomenen die in de cloud hype lijken mee te liften zijn Het Nieuwe Werken en Bring Your Own Device. Wordt de ondersteuning van een veelheid van vaste en mobiele platforms een belangrijk risico bij cloud computing of niet? De toekomst zal het leren.

## Conclusie

Voor de testmanager is een belangrijke rol weggelegd bij de invoering van cloud computing. De testmanager moet al in actie komen bij de selectie van cloud services. Maar ook na de implementatie ervan loopt zijn rol door tot in de operatie van cloud services. Centraal staan daarbij de risico's en (test)gerelateerde maatregelen om die risico's te beheersen. «

**'Een wild en pril idee, maar tot dusver is het niet weggehoond.'**