Netzwerktest mittels Lasttestwerkzeug: Definierte Netzwerklast bewusst herbeiführen

Standardisierte Vorgehensweisen, mit denen Gewissheit über die Leistungsfähigkeit eines Softwareproduktes erlangt werden kann, sind in der Softwarequalitätssicherung längst etabliert und werden nur unzureichend für das Bewusstsein über das eigene Firmennetzwerk eingesetzt. Wie verschaffen Sie sich Gewissheit über die Leistungsfähigkeit Ihres Netzwerkes? Wie stellen Sie Fehler in der Netzwerkstruktur nach und welche etablierten Techniken können Sie hierbei nutzen? Diesem Artikel liegt ein Zwischenfall während einer Pressekonferenz eines norddeutschen Großkonzerns zugrunde.

Der klassische Lasttest im Softwarelebenszyklus

Selbst die beste Software ist nicht nutzbar, wenn jeder Klick zur Mittagszeit einen nicht enden wollenden Ladebalken hervorruft oder eine Vielzahl von Angestellten den Videostream der konzerneigenen Pressekonferenz mitverfolgt und den Server verlangsamt. Grundlegender Zweck des Lasttests ist es daher, sich ein Bewusstsein darüber zu verschaffen, wie sich die Leistungsfähigkeit eines Softwareproduktes unter der zunehmenden Arbeitslast von gleichzeitigen Benutzern ändert und ob das Testobjekt letztlich der erwarteten Anzahl von Anwendern standhält. Kenngrößen, die diese Performanz definieren, sollten in der Softwareentwicklung in Form von nicht-funktionalen Produktanforderungen definiert und schon in der Konzeptionsphase berücksichtigt werden.

Der Lasttester setzt bei seiner Arbeit die störungsfreie Funktionalität der Netzwerkinfrastruktur voraus, da er seinen Fokus zumeist auf die Überprüfung der Qualitätsmerkmale der zu testenden Applikation unter verschiedenen Lastzuständen legt. Vereinfacht gesagt setzt er voraus, dass die Netzwerkinfrastruktur niemals die Schwachstelle seiner Messungen darstellt *(weiterführende Information: [HPLR, LTUe])*.

Die Werkzeuge des klassischen Lasttests

Für Informationssysteme, die auf eine Vielzahl von parallel arbeitenden Nutzern ausgelegt sind, ist es nicht wirtschaftlich möglich, das Lastverhalten manuell zu ermitteln. Eine effiziente Alternative bietet an dieser Stelle eine Palette von kommerziellen und nicht-kommerziellen Lasttesttools. Diese Tools können viele gleichzeitig arbeitende Anwender simulieren und den eigentlichen Testprozess automatisiert durchführen. Hierfür bieten gängige Werkzeuge die Möglichkeit, den Netzwerkverkehr zwischen Client- und Server-Anwendung aufzuzeichnen und nach individueller Anpassung als Skript zu hinterlegen. Das erstellte Skript kann während der eigentlichen Testdurchführung über eine variable Dauer hinweg beliebig oft parallel ausgeführt werden.

Jede einzelne der parallel arbeitenden Skriptausführungen innerhalb des Tests simuliert näherungsweise einen realen Anwender. Der Erfolg dieser Annäherung hängt hauptsächlich von der Komplexität der zu testenden Anwendung sowie dem Detailgrad der Funktionalitäten ab, die in Form von repräsentativen Geschäftsprozessen abgebildet werden sollen. Jeder dieser ausgewählten Geschäftsprozesse ist wiederum Grundlage für das exakt umzusetzende Lasttestskript. Wie bereits angesprochen, kann die Komplexität der Geschäftsprozesse beliebig verfeinert werden. Ein ausreichender Realitätsgrad kann aber oft bereits mit der Abbildung von 5 bis 20 Klick-Aktionen innerhalb eines Geschäftsprozesses erreicht werden *(weiterführende Information: [Kle13, Koo08, Wiki-a])*.

Netzwerktechnik, das Nervensystem der IT

Das Netzwerk versorgt den Computer mit Daten aus aller Welt. Bleiben diese Daten aus, so ist nicht nur eine Software, sondern zumeist das gesamte System schlagartig nicht mehr nutzbar. Erreichen die Daten verzögert ihr Ziel, schlägt sich dies zwangsläufig in Wartezeiten innerhalb der Anwendung nieder, die diese Daten benötigt.

Netzwerke, die die Infrastruktur moderner Applikationen bereitstellen, müssen proportional zu den steigenden Anforderungen an die Anwendungen selbst wachsen. Je größer das Netzwerk ist, desto größer wird auch das Risiko von Konfigurationsfehlern oder Hardwareausfällen *(weiterführende Information: [Wiki-b])*.

Wie schnell und zuverlässig ist Ihr Netzwerk denn nun?

Eine entsprechende Fachabteilung wird die Frage nach der Geschwindigkeit mit der Angabe eines maximalen Gesamtdurchsatzes im Bereich von 100 Megabit pro Sekunde bis hin zu 100 Gigabit pro Sekunde beantworten. Diese Angabe stellt im Grunde keine Geschwindigkeit dar, sondern beziffert eine Datenmenge, die pro Sekunde ungehindert vom Netzwerk übertragen werden kann. Verzögerungen durch das Netzwerk wird der Anwender erst spüren, wenn die abzuwickelnde Datenmenge den maximalen Gesamtdurchsatz einer betroffenen Netzwerkstrecke übersteigt. Ähnlich wie bei einem Stau auf der Autobahn verlangsamt sich die effektive Übertragungsgeschwindigkeit schlagartig auf einen sehr geringen Wert.

Um einschätzen zu können, wann eine Überschreitung des maximalen Datendurchsatzes droht, müssen zunächst folgende Fragen beantwortet werden:

- Wie viel Netzwerklast erzeugt ein durchschnittlicher Anwender?
- Wie viele potenziell gleichzeitige Anwender gibt es und welche Tätigkeiten führen sie aus?
- Wie häufig und wie schnell führen die Anwender diese Tätigkeiten aus?
- Welche Datenmengen werden bei den Aktivitäten über das Netzwerk versendet oder empfangen?

Abb. 1: Beispielhafter Verkehrsfluss zwischen Hamburg, Bremen, Bielefeld und Berlin. (Kartendaten © GeoBasis-DE/BKG [©2009] Google).

- Wo in der Netzwerktopologie befinden sich bei den einzelnen Tätigkeiten Quelle und Ziel der Kommunikation und welche Netzwerkstrecken werden dabei potenziell belastet?

Die Angabe des maximalen Datendurchsatzes wird nur anhand von Datenblättern der verwendeten Netzwerkkomponenten bestimmt und in der Regel nie in der konkreten Konfiguration und im realen Umfeld nachgewiesen. Im Gegensatz zur getesteten Software fehlt hier also jegliche Gewissheit über die real vorhandene Leistungsfähigkeit.

Lasttest gleich Netzwerktest?

Besitzen wir bereits ein Skript, welches einen Anwender simuliert, der mit einer beliebigen Applikation arbeitet, so kann dieses auch für einen Test der Netzwerkstruktur verwendet werden. Es sei an dieser Stelle jedoch angemerkt, dass es prinzipiell erst einmal egal ist, welches Skript beziehungsweise welche Anwendung letztlich den Netzwerkverkehr hervorruft, den wir erzeugen wollen. Eine erste Grundlage zur Erstellung des Netzwerktests ist jedoch vorhanden und dient zudem als Basis für die Berechnung des erzeugten Datendurchsatzes eines realen Anwenders bei seiner täglichen Arbeit.

Den entscheidenden Mehrwert liefern an dieser Stelle das im Lasttest verwendete Werkzeug und die beteiligte Infrastruktur. Kritische Pfade des Netzwerkes, auf denen sich die Datenströme mehrerer Anwender bündeln, sind für diese Datenmengen ausgelegt und können meist nicht von einem einzigen Computer als Quelle vollständig ausgelastet werden. Die Infrastruktur des Lasttesttools liefert eine einfache Möglichkeit, mehrere Computer gleichzeitig dazu anzuweisen, das Lasttestskript auszuführen und damit die Netzwerktopologie zu belasten.

Die als Lastgenerator bezeichneten Systeme können durch physikalisches Umstecken des Netzwerkanschlusses oder durch geeignete Konfigurationsänderungen der Netzwerkparameter flexibel an beliebigen Knotenpunkten des Netzwerks eingebracht werden. Die Nutzung dieser mitgelieferten Infrastruktur bietet eine kosteneffiziente Möglichkeit, gezielt die relevanten Netzwerkstrecken zu prüfen. Da Lastgeneratoren ausschließlich auf diese Aufgabe optimiert sind, kann jeder einzelne eine große Anzahl von realen Benutzern simulieren. Die konkrete Anzahl, die pro Lastgenerator simuliert werden kann, hängt hierbei sehr stark vom verwendeten Werkzeug sowie der Komplexität der Lasttestskripte ab.

Das eigentliche Lasttesttool, welches den Testausführungsprozess steuert, protokolliert parallel die relevanten Messwerte und bereitet diese in den meisten Fällen angemessen auf.

Abstrahierter Test

Analog zum Straßennetz könnte man sagen, dass ein Problem auf der A2 festgestellt wurde: Zwischen Kreuz Hannover und Berlin treten im Berufsverkehr Verkehrsstörungen auf (**siehe Abbildung 1**). Man würde beispielsweise Hamburg, Bremen und Bielefeld als Lastgeneratoren verwenden und Fahrzeuge von dort koordiniert nach Berlin schicken.

Organisatorisch muss dabei beachtet werden, dass beispielsweise die Strecke von Hamburg über die A24 in Richtung Berlin gesperrt wird, um die definierte Menge der Fahrzeuge – oder Netzwerkpakete – in definierter Zeit auf den zu analysierenden Streckenabschnitt schicken zu können. Dann können Start- und Ankunftszeiten protokolliert und der betroffene Streckenabschnitt reproduzierbar in die problematische Situation versetzt werden. Danach findet eine Analyse und gegebenenfalls

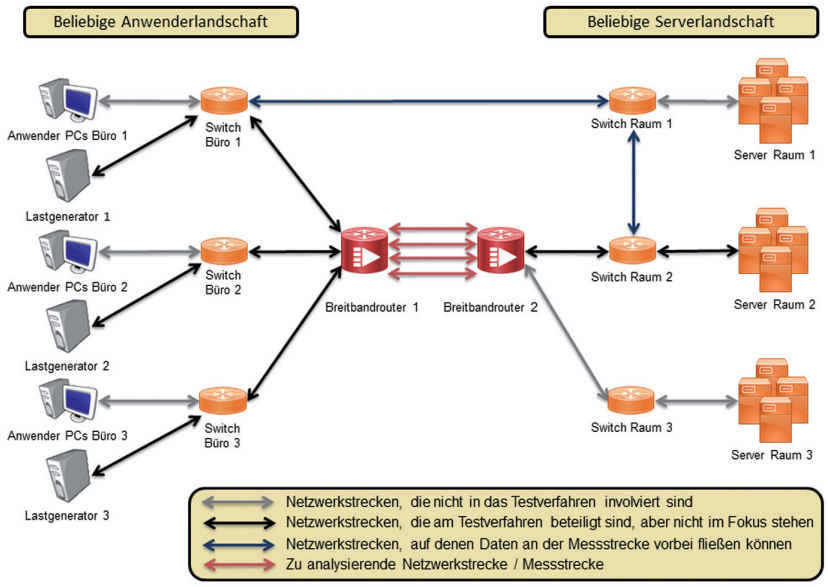

Abb. 2: Vereinfachte Routingstruktur innerhalb eines Netzwerkes unter Verwendung der Cisco-Darstellung. These materials have been reproduced by Sogeti Deutschland GmbH with the permission of Cisco Systems Inc. © CISCO SYSTEMS, INC. ALL RIGHTS RESERVED.

Änderung am Streckenabschnitt statt. Die durchgeführten Änderungen können durch einen erneuten Test validiert werden.

Systematischer Testaufbau

Um die eingangs angesprochene, während der Pressekonferenz eingetretene, Verlangsamung nachzustellen, wurden die als ursächlich verantwortlich identifizierten Netzwerkstrecken durch konfigurative Anpassungen und geschickte Auswahl der Standorte der Lastgeneratoren und des Zielservers unter Last gesetzt.

Die für die Störung verantwortliche Netzwerkstrecke sei in diesem Testaufbau die Verbindung zwischen zwei Breitbandroutern. Sie verbinden das Teilnetzwerk, in dem sich die Anwender befinden, mit dem Teilnetzwerk, in dem sich die Videostream-Server befinden. Da die Anwender bei ihrer täglichen Arbeit eine hohe Datenlast auf dieser Schnittstelle erzeugen, wurde auch in **Abbildung 2** eine leistungsfähige Anbindung berücksichtigt. Im konkreten Testfall sorgen vier Glasfaserleitungen mit einem maximalen Datendurchsatz von jeweils 10 GBit/s für einen reibungslosen Datenaustausch.

Voraussetzung für die simple Addition der Einzelleitung hin zum Gesamtdatendurchsatz dieser Anbindung ist eine gleichmäßige Verteilung der einzelnen Anfragen auf alle vier zur Verfügung stehenden Glasfaserleitungen. Die Systematik im aktuellen Fall lässt es nicht zu, die Kommunikation einer einzelnen Quelle auf mehrere Glasfaserleitungen aufzuteilen, daher werden mehrere Lastquellen verwendet, um den Datenstrom zu erzeugen.

Die Überwachung der Netzwerkparameter ergibt, dass die Übertragung der einzelnen Netzwerkpakete immer länger dauert, obwohl das theoretisch erwartete Limit von 40 GBit/s nicht vollständig ausgelastet ist. Diese Netzwerkstrecke soll daher gezielt und kontrolliert auf ihre Leistungsparameter hin überprüft werden.

Lasttransport

Die einzelnen Teilnetzwerke, die in der vorliegenden Situation betroffen sind, werden in **Abbildung 2** auf die relevanten Teile „Anwenderlandschaft“ und „Serverlandschaft“ reduziert. Die Topologie und die damit verbundenen Routingmöglichkeiten werden in der Realität wesentlich komplexer sein und in jedem Fall eine detaillierte Analyse durch die Fachabteilung notwendig machen.

Bedingung für einen verwertbaren Test ist es, wie im abstrakten Beispiel mit technischen Mitteln dafür zu sorgen, dass die von den Lastgeneratoren gesendeten Daten vollständig über die zu testende Netzwerkstrecke abgewickelt werden. Die Verwendung der in **Abbildung 2** blau dargestellten Netzwerkstrecke würde das Testergebnis verfälschen.

An dieser Stelle sei nur grundlegend darauf hingewiesen, dass die Berücksichtigung der verfügbaren Netzwerkstrecken eine größere Herausforderung darstellt, als deren Deaktivierung. Der Netzwerkabteilung stehen hierfür Routingtabellen, physikalisches Trennen sowie die Nutzung von Firewalls zur Verfügung.

```
web_custom_request(
  "Beispielbild von Beispielseite.org anfordern"),
  "URL=www.Beispielseite.org/Beispielbild.jpg",
  "Method=GET",
  "TargetFrame=",
  "Resource=0",
  "RecContentType=image/jpeg",
  "Mode=HTML",
  "EncType=application/x-www-form-urlencoded; charset=UTF-8",
LAST);
```

Listing 1: Beispielhaftes C-Skript für eine Netzwerkanfrage an einen Webserver. Erstellt mit HP Virtual User Generator (siehe [HPLR]).

Lastziel

Essenziell für dieses Verfahren ist ein Lastziel (Server), welches beliebig häufig parallel und über variable Zeit hinweg auf die Anfrage eines Clients reagieren und diese beantworten kann. Bedingung ist es natürlich auch, dass dieses System die Anfragen so schnell beantwortet, dass der gewünschte Gesamtdatendurchsatz erreicht wird. Ein Webserver, dessen primäre Aufgabe es ist, Webseiten zu liefern, wird so eingerichtet, dass er ein zuvor definiertes Bild an den Anfragenden zurücksendet.

Die Arbeitslast kann mittels Loadbalancing-Technologie auf mehrere Webserver aufgeteilt werden, sodass das „Lastziel“ theoretisch mehr Daten liefern könnte, als das Netzwerk physikalisch übertragen kann.

Lastquelle

Der Begriff „Lastquelle“ wird hierbei im übertragenen Sinn für den Client verwendet, der das vom „Lastziel“ angebotene Bild abruft. Hat dieses Bild eine definierte Größe und wird festgelegt, wie oft es parallel abgerufen wird, so kann einfach berechnet werden, wie viel Last auf der Schnittstelle zwischen den Netzwerken abgewickelt werden muss. Ein kleiner Aufschlag von 5 Prozent sollte für die Anfrage des Bildes selbst und Randeffekte mit eingerechnet werden. Dieser Wert dient aber nur für die grobe Konzeptionierung des Tests. Die Anzahl der virtuellen Anwender, beziehungsweise in Folge die erzeugte Gesamtlast, kann im Lasttesttool überwacht und gegebenenfalls durch den Start weiterer Instanzen der Lastgeneratoren nachgesteuert werden.

Die Infrastruktur der Lastgeneratoren und ihrer Randsysteme wurde im konkreten Fall mit Produkten aus dem HP Application Lifecycle Management erstellt. Es gibt am Markt jedoch einige Konkurrenzprodukte, die vergleichbare Anforderungen erfüllen.

Listing 1 zeigt die Einfachheit eines für diese Zwecke notwendigen Lasttestskriptes.

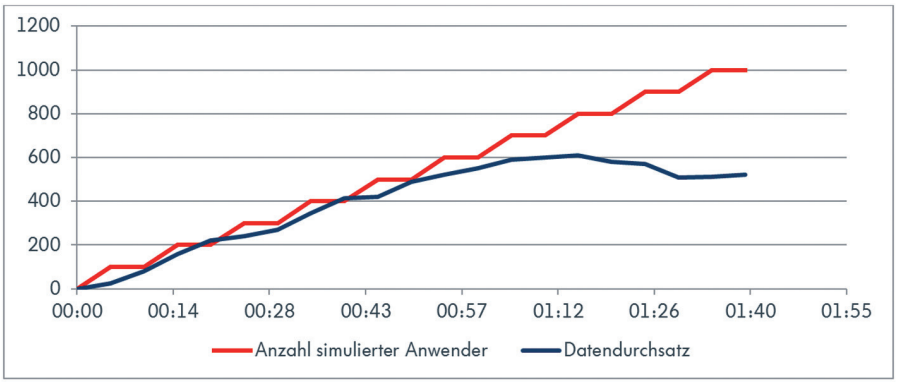

Abb. 3: Stagnierender Datendurchsatz trotz steigender Anzahl simulierter Benutzer.

Einbindung in den regulären Testprozess

Gerade dann, wenn wie im vorliegenden Fall bereits der Fehlerfall eingetreten ist, lässt sich dieses Verfahren natürlich nicht mehr in den regulären Testbetrieb einbinden, sondern muss in einer der wenigen Phasen von geringer Netzwerkauslastung durchgeführt werden. Die Risiken von Ausfällen in der Nacht oder an Wochenenden müssen individuell ermittelt und berücksichtigt werden.

Um dieser Zeitnot nicht ausgesetzt sein zu müssen, kann dieses Verfahren auch vorbeugend während eventuell bereits geplanter Wartungszeiträume innerhalb der Serverlandschaft durchgeführt werden.

Testdurchführung

Über die Steuerung des Lasttesttools kann ein Szenario erstellt werden, welches die Anzahl der User, die das in **Listing 1** beschriebene Skript parallel ausführen, kontinuierlich erhöht. Es empfiehlt sich, die Anzahl bis zu einem unkritischen Niveau per Szenario zu regeln und dann gegebenenfalls während der Testlaufzeit manuell nachzusteuern. Der durch den Prozess erzeugte Gesamtdatendurchsatz wird von der Mehrheit der Lasttesttools in einem Livemonitor dargestellt. Parallel hierzu kann das Netzwerkteam die Monitore überwachen, die auch während der Pressekonferenz eine Verlangsamung des Paketdurchsatzes angezeigt haben.

Stagniert der Datendurchsatz trotz zunehmender Anzahl von simulierten Anwendern (**siehe Abbildung 3**), so wurde der aktuell angezeigte Wert als maximaler Datendurchsatz bestimmt. Liegt dieser unter dem erwarteten Wert, kann mittels Ausschlussverfahren die Ursache für das unerwartete Bottleneck ermittelt werden.

Ursachen für eine Differenz zwischen gemessenem und berechnetem Datendurchsatz können sein:

- Ziel unterdimensioniert: kann mittels (Monitoring) von Werten wie CPU, Speicher, Netzwerkdurchsatz usw. nachgewiesen werden. Eine Erweiterung des Zielsystems schafft Abhilfe.
- Quelle unterdimensioniert: die Systemparameter werden von einigen Lasttesttools automatisch überwacht. Sollte dies nicht der Fall sein, so kann diese Überwachung mit Betriebssystem-Bordmitteln selbst eingerichtet werden. Eine Zuschaltung von weiteren Lastgeneratoren schafft auch hier schnelle Abhilfe.
- Ein realer Fehler im Netzwerk konnte nachgewiesen werden: Da der Test beliebig wiederholbar ist, hat das Netzwerkteam nun die Möglichkeit, den Fehler einzugrenzen und zu beheben. Während des Tests nach der Pressekonferenz wurden die ersten Punkte ausgeschlossen und das Phänomen wurde auch in den Monitoren des Netzwerkteams erneut sichtbar. Eine gezieltere Analyse der Datenströme auf allen verfügbaren Glasfaserleitungen führte letztlich zu der Erkenntnis, dass ein kleiner Fehler in der Konfiguration dazu geführt hat, dass nur zwei der vier Glasfaserleitungen genutzt wurden. Dieser Fehler hätte ohne das vorgestellte Verfahren, wenn überhaupt, nur mit höherem Aufwand nachgewiesen werden können, da er erst ab einem gewissen Datendurchsatz sichtbar wird. Erst die Pressekonferenz und schließlich der erstellte Test konnten die Voraussetzung schaffen, um diesen Fehler zu reproduzieren.

Literatur & Links

[HPLR] Hewlett Packard, LoadRunner, siehe: `www8.hp.com/us/en/software-solutions/loadrunner-load-testing/index.html`

[Kle13] B. Klemm, Application Performance Testing: A Universal Performance Testing Methodology, Kindle E-Book, 2013

[Koo08] T. Koomen, L. van der Aalst, B. Broekman, M. Vroon, TMap Next, dpunkt.verlag, 2008

[LTUe] International Testing-Board, Lasttest Tools Übersicht, siehe: `www.testing-board.com/lasttest-tools-uebersicht/`

[Wiki-a] Wikipedia, Lasttest, siehe: `de.wikipedia.org/wiki/Lasttest_(Computer)`

[Wiki-b] Wikipedia, Rechnernetz, siehe: `de.wikipedia.org/wiki/Rechnernetz`

Fazit

Wie in vielen Bereichen der Qualitätssicherung bekannt, muss natürlich auch für diese Vorgehensweise eine umfassende Kosten/Nutzen-Abschätzung erstellt werden, die letztlich vom Risiko eines Ausfalls und der drohenden Beeinträchtigung des Netzwerkes beziehungsweise der resultierenden Auswirkungen abhängig ist. Dem Ausfall der Übertragung des Streams zum Server des TV-Senders, der die Pressekonferenz überträgt, wäre sicherlich eine größere Auswirkung zuzuschreiben als einem leichten Ruckeln des Videostreams an den Arbeitsplätzen der Mitarbeiter.

Spätestens im Fehlerfall kann ein guter Kontakt zwischen Experten aus dem Last- und Performanzumfeld und dem für das Netzwerk zuständigen Fachbereich nur Vorteile bringen. ||

Der Autor

|| Marcel Jodorf
(marcel.jodorf@sogeti.de)
ist als Consultant bei der Sogeti Deutschland GmbH im Bereich Softwarequalitätssicherung für einen norddeutschen Großkonzern tätig. Seine Aufgabenbereiche umfassen Beratung, Konzeptionierung, Durchführung sowie Auswertung von Last- und Performanztests für verschiedene Softwareprojekte.

OBJEKTspektrum ist eine Fachpublikation des Verlags:
SIGS DATACOM GmbH · Lindlaustraße 2c · 53842 Troisdorf
Tel.: 02241/2341-100 · Fax: 02241/2341-199
E-mail: info@sigs-datacom.de
www.objektspektrum.de
www.sigs.de/publications/aboservice.htm
SIGS DATACOM
FACHINFORMATIONEN FÜR IT-PROFESSIONALS