FOKUS: OPEN SOURCE

Die Einfachheit trügt

Dank Cloud und fertigen Software-Komponenten ist eine Geschäftsidee rasch und günstig umgesetzt. Doch was von aussen einfach aussieht, wird hinter den Kulissen immer komplexer. Wer kümmert sich z. B. um die Sicherheit und Verfügbarkeit der vielen Komponenten sowie um die Anwendung nach dem Go-Live?

→ VON KASPAR GEISER

Fast in jedem Geschäftsfeld steht heute eine Vielzahl von Open-Source-Anwendungen zur Verfügung. Das gilt für E-Mail, Chat und Dateiaustausch genauso wie für CMS-Systeme. Selbst für Smartphone-Apps gibt es Frameworks für wenig Entgelt. Software-as-a-Service-Modelle (SaaS) gehen noch weiter: Der Anwender muss keine Software mehr installieren, sondern nutzt die Anwendung als Dienst im Web. Wer solche fertigen Komponenten verwendet, kommt leicht in Versuchung, auf die Betriebs- und Supportorganisation zu verzichten. Eine Anwendung im Internet ohne Hoster zu betreiben, schien noch vor wenigen Jahren undenkbar, heute ist dies theoretisch möglich. Die Praxis aber zeigt: Der Hoster ist wichtiger denn je, als Aufsichtsinstanz und Berater für einen sicheren Betrieb.

AUSWAHL: WENIGER IST EINFACHER

Es beginnt bereits bei der Planung einer Anwendung. Tools sind im Netz schnell evaluiert, Konfiguration und Programmierung folgen sofort. Doch beim Wechsel von den Testservern in die Cloud oder auf die produktiven Systeme eines Hosters drohen die ersten Gefahren. Woher stammt die verwendete Software? Wie ermittle ich, dass sie sauber ist? Welche Abhängigkeiten zu anderer Software gibt es? Wer diese zentralen Fragen nicht stellt, riskiert Ausfälle oder Sicherheitslücken.

In dieser Phase besteht die Aufgabe des Hosters darin, Infrastruktur und Anwendung aufeinander abzustimmen und zu prüfen, ob alle Komponenten aus vertrauten und geprüften Quellen stammen. Weniger ist dabei oft mehr. Wer nur die wirklich benötigten Komponenten wie Webserver und Datenbank installiert, hat den besseren Überblick. Komplette Managementframeworks für die Administration vereinfachen die Aufgabe zwar vordergründig, verkomplizieren aber das Sicherheitsmanagement.

Zur Planungsphase gehören weitere Aufgaben, die von Anwendern nur schwer übernommen werden können. Hoster eruieren in dieser Phase die Anforderungen an die Verfügbarkeit und setzen sie um. Dazu gehört etwa, eine Webanwendung durch Loadbalancer hochverfügbar zu machen. Weiter soll zwischen Hoster, Entwickler und dem Business die Segregation of Duty bestimmt werden. Es gilt festzulegen, wer welches Recht in einer Produktion hat: Darf nur der Hoster ein Release in der Produktion live schalten oder kann dies ein Entwickler selbstständig tun? Nicht zuletzt überwacht der Hoster die Umsetzung funktionierender Backups und Restore-Prozeduren sämtlicher Anwendungen.

BETRIEB: PATCHES UND UPGRADES

Kaum ist eine Anwendung in Betrieb, stehen Patches und Upgrades an. Allein WordPress gibt pro Jahr gut zwei Dutzend Releases heraus. Dasselbe gilt für MySQL, das oft als Datenbank für Open-Source-Anwendungen genutzt wird. Da stellt sich die Frage, ob diese Patches wirklich alle installiert werden müssen – und wie das überhaupt geht. Ausserdem muss immer klar sein, wie ein Fallback aussieht.

«Die Rolle des Hosters hat sich in den letzten Jahren vom Infrastrukturanbieter zum Berater gewandelt»

Kaspar Geiser

Wer sich eine Anwendung aus Open-Source-Komponenten oder Frameworks zusammenbaut, ist in diesen Punkten stark gefordert. Mit jeder Komponente tauchen mehr Fragen auf. Beispiel jQuery: Welche Folgen hat eine Aktualisierung? Unterstützt die Anwendung dann die gängigsten Internet-Explorer-Versionen noch? Und was sind die Risiken, wenn jQuery in der älteren Version weiterbenutzt wird? Weitere Hürden können Applikationsserver wie Jboss oder Tomcat sein. Müssen diese ebenfalls aktualisiert werden oder bloss die Anwendungen, die darauf betrieben werden? Solche Fragen können nur Leute beantworten, die ein System von der Hardware über die schützenden Firewalls bis zu den Anwendungsfällen kennen.

VORSICHT, ABHÄNGIGKEITEN!

Viele Projekte starten auf der grünen Wiese. Dies bringt zu Beginn grosse Freiheiten – doch die Anwendungen stehen nur am Anfang allein da. Kaum sind sie in Betrieb, entstehen Abhängigkeiten. Bezieht eine Anwendung Daten aus einem ERP, macht eine Umstellung im ERP vielleicht eine Anpassung der Webschnittstelle nötig. Die Wahl von Betriebssystem und Anwendung kann unerwartete Abhängigkeiten mit sich bringen. So haben Sicherheitslücken wie Poodle und Logjam dazu geführt, dass neben den Webservern auch die darunterliegenden Betriebssysteme aktualisiert werden mussten. Eine andere Falle: Manche Patches machen Anwendungen für gewisse Browser inkompatibel, was Anwender vergrault.

DEVOPS – AUF EIGENE FAUST

Der Trend zum Self Service führt dazu, dass in kleinen und grossen Projekten auf den Hoster verzichtet wird. Viele seiner Aufgaben stehen dadurch in der direkten Verantwortung des Anwendungseigners. Das gilt zum Beispiel für DevOps-Prozesse. Sie werden zwar vereinfacht, wenn das Hosting als Stufe zwischen Business und Betrieb wegfällt. Zugleich erhöht sich dabei aber die Verantwortung für die Entwickler. Denn bei DevOps-Prozessen gelangen Konfigurationen ohne Vier-Augen-Prinzip in die Produktion. Um auch nach dem Go-Live seine Anwendung sicher betreiben zu können, sind neben den klassischen ITIL-Disziplinen auch Kenntnisse im Bereich Schwachstellenanalyse notwendig. Gerade im Open-Source-Umfeld ist das Eruieren von Schwachstellen aufwendig, da der Quellcode nicht von einem bekannten Hersteller, sondern aus diversen Quellen stammt.

VOM BETREIBER ZUM BERATER

Die Rolle des Hosters hat sich in den letzten Jahren vom Infrastrukturanbieter zum Berater gewandelt. Heute bestimmt er bereits zu Beginn eines Projekts Verfügbarkeits- und Sicherheitsanforderungen einer Architektur. Er durchleuchtet die von einer Anwendung verwendeten Software-Komponenten und macht die Entwickler auf Risiken im Zusammenhang mit dem Ursprung der Software aufmerksam.

Über den gesamten Lebenszyklus einer Anwendung soll der Hoster zudem bekannte Gefahren und Schwächen einer Anwendung kennen und diese regelmässig überprüfen. Er muss erkannte Schwachstellen plausibel aufzeigen und Möglichkeiten zu deren Behebung vorschlagen. Da eine Anwendung nicht bloss aus einem Serverteil besteht, muss ein Hoster zudem über die Gefahren auf Anwenderseite im Bilde sein.

FAZIT: MEHR KNOW-HOW ERFORDERLICH

Open Source, Gratis-Software und SaaS können Projekte beschleunigen, sie machen sie aber nur scheinbar einfacher. Self Service kann aus Anwendungen komplexe Gefüge machen, sodass Anwender Schwachstellen und Verantwortlichkeiten allein kaum mehr überblicken. Hosting wird dadurch zu einer neuen Aufgabe, Betreiber müssen alles im Auge behalten: Vom Hardware-Einsatz bis zum Patching von Cloud-Anwendungen. Hoster werden zu Generalunternehmern, sie müssen Netzwerkspezialisten, Applikationsingenieure, Sicherheitsspezialisten, Projektmanager, Kommunikatoren und Berater in einem sein.

Das reicht aber noch nicht. Denn auch als Generalunternehmer kann der Hoster nicht alle Fragen allein beantworten. Es braucht eine enge Zusammenarbeit zwischen Business, Entwicklern und dem Betrieb. Ein regelmässiger Informationsaustausch ist Pflicht. Geografische Nähe und eine gemeinsame Sprache werden dabei wieder wichtiger. ←

Kaspar Geiser ist Geschäftsführer der Aspectra AG
→ **www.aspectra.ch**

BILD: ISTOCKPHOTO.COM/CJP