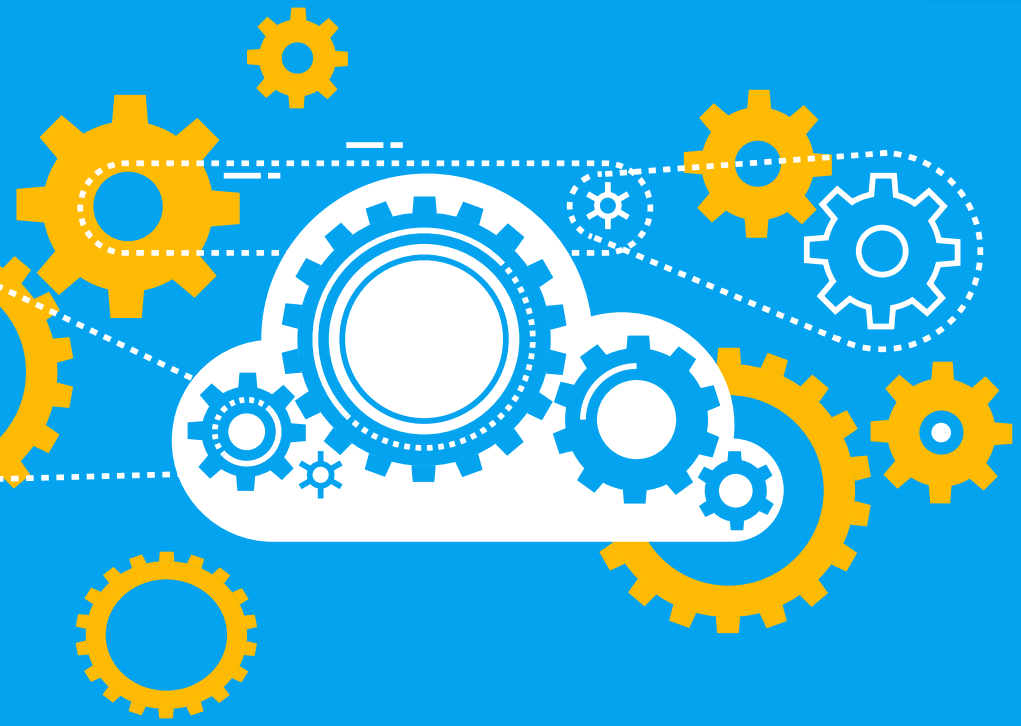
RAPORT



Jak szybko zbudować aplikację

z gotowych narzędzi w chmurze

ITWIZ
TO CO ISTOTNE W IT

Microsoft

ADAM JADCZAK

Bank Millennium

uruchomił platformę goodie z wykorzystaniem narzędzi w chmurze Microsoft Azure

Bank Millennium uruchomił goodie – otwartą dla wszystkich platformę zakupową, która powstała na zasadzie wewnętrznego start-upu. To nieszablonowe rozwiązanie w skali polskiego rynku usług bankowych. W nowym projekcie bank wykorzystuje elementy sztucznej inteligencji, Machine Learning, rozpoznawanie obrazów czy geolokalizację. W celu szybkiego udostępnienia aplikacji klientom, zespół goodie zdecydował się nie tworzyć wszystkich narzędzi od podstaw, a skorzystać z niektórych gotowych komponentów dostępnych w chmurze Microsoft Azure.

Platforma goodie zbiera informacje na temat zniżek, promocji i nowości zakupowych w centrach handlowych, sklepach internetowych i stacjonarnych oraz restauracjach. Użytkownik znajdzie w aplikacji inspiracje zakupowe oraz oferty dopasowane do preferencji i miejsca, w którym się znajduje, np. najbliższej galerii handlowej. Platforma goodie to także kanał komunikacji sklepów i centrów handlowych z klientami. Ma im pozwolić dotrzeć do nowych grup konsumentów, oferować zróżnicowane zniżki i promocje.

FINTECH ROZWIJANY WEWNĄTRZ BANKU

Dotychczas – w procesie innowacji, wykorzystania nowych technologii – banki albo kupowały spółki FinTech, albo starały się z nimi współpracować przy określonych projektach. Bank Millennium zdecydował się na zbudowanie własnego start-upu, bazując na zgromadzonych doświadczeniach. *„Zbierane były one np. na podstawie tego, w jaki sposób klienci banku korzystają z aplikacji mobilnej. Jesteśmy też jednym z pierwszych banków, które wykorzystywały rozszerzoną rzeczywistość czy narzędzia Voice Recognition do identyfikacji użytkowników naszych serwisów. Teraz eksperymentujemy z chat botami"* – mówi Ricardo Campos, dyrektor Departamentu Bankowości Elektronicznej w Banku Millennium i jednocześnie prezes spółki Millennium Goodie.

Dlaczego w Banku Millennium powstał pomysł stworzenia platformy goodie? *„Świat finansów się zmienia. Motorem zmian są rosnące oczekiwania klientów, dynamiczne otoczenie gospodarcze i nieustanny rozwój technologii. Dlatego m.in. jednym ze strategicznych filarów rozwoju banku stała się innowacyjność i nowe technologie. Realizacja tej strategii na tak konkurencyjnym rynku – w otoczeniu firm z sektora FinTech, IT i firm telekomunikacyjnych – wymaga od nas nie tylko stałego rozwijania internetowych i mobilnych usług bankowych, lecz także otwarcia na nieszablonowe rozwiązania. W związku z tym stworzyliśmy wewnątrz banku multidyscyplinarny zespół ludzi, współpracujących ze sobą na zasadzie start-upu. Tak powstało goodie, cyfrowa platforma oferująca ekosystem, w którym sprzedawcy, klienci i bank – poprzez wzajemne interakcje – tworzą wartości dla wszystkich stron. Jest wiele scenariuszy rozwoju branży. Być może w najbliższej przyszłości banki staną się integratorami usług niezwiązanych z bankowością i w ten sposób znajdą miejsce w cyfrowej rzeczywistości klienta"* – dodaje Ricardo Campos.

PLATFORMA NIEOGRANICZONEGO KONTAKTU Z KLIENTEM

Jak tłumaczą przedstawiciele Banku Millennium, jeśli myślimy o bankowości, to zwykle w kontekście pożyczenia lub zdeponowania pieniędzy. Bank musi być w tym momencie jak

najbliżej klienta. Dlatego instytucje te najpierw otwierały oddziały, następnie tworzyły sieci bankomatów, a obecnie inwestują w kanały internetowe i bankowość mobilną. *Nasi klienci coraz więcej czasu spędzają w internecie. A nie jest to dziś przestrzeń, gdzie firmy mają swobodny dostęp do swoich klientów. Ograniczają go właściciele platform internetowych. Mam dostęp do klienta, ale tylko wtedy, gdy za to zapłacę"* – twierdzi Ricardo Campos. *„Z drugiej strony na zarezerwowany dotąd dla banków rynek wchodzą wielkie firmy technologiczne. Nie tworzą jednak własnych banków, a jedynie udostępniają użytkownikom swoje usługi, np. systemy płatności. Jednocześnie banki – zgodnie m.in. z dyrektywą PSD2 – muszą otwierać się na podmioty zewnętrzne i udostępniać im dane. Przypomina to historię operatorów telekomunikacyjnych, którzy też musieli udostępniać swoje sieci, ponosząc przy tym koszty ich rozbudowy i utrzymania"* – dodaje.

Stąd pomysł w Banku Millennium na platformę goodie, która ma być maksymalnie otwartym ekosystemem dla klientów i merchantów. Przedstawiciele Banku Millennium pokazują – jako przykład skutecznej konkurencji z dużymi, międzynarodowymi graczami – uruchomienie w Polsce platformy BLIK.

Cały projekt goodie – od opracowania koncepcji platformy aż do udostępnienia jej klientom i merchantom – **trwał krócej niż rok. Skrócenie czasu,** jaki poświęciliśmy na wprowadzenie naszej koncpecji w życie, **uzyskaliśmy m.in. dzięki wykorzystaniu gotowych elementów funkcjonalnych w chmurze Microsoft Azure.** Zapewniło to także odpowiednią skalowalność infrastruktury.

Ricardo Campos, dyrektor Departamentu Bankowości Elektronicznej w Banku Millennium i jednocześnie prezes spółki Millennium Goodie

W NIECAŁY ROK OD POMYSŁU DO GOTOWEGO PRODUKTU

Bank Millennium uruchomił projekt goodie w 2016 roku. Wersja beta serwisu i aplikacji goodie udostępniona została do testów już w grudniu ubiegłego roku. *„Cały projekt – od opracowania koncepcji platformy aż do udostępnienia jej klientom i merchantom – trwał krócej niż rok. Skrócenie czasu, jaki poświęciliśmy na przygotowanie samego rozwiązania, uzyskaliśmy dzięki wykorzystaniu w realizacji naszej koncepcji niektórych gotowych elementów funkcjonalnych i wbudowanych procesów w chmurze Azure. Zapewniło to również odpowiednią skalowalność infrastruktury"* – tłumaczy Ricardo Campos.

Dlatego też w tworzeniu goodie aktywny udział wziął Microsoft. *„Naszym zadaniem była pomoc w przełożeniu pomysłu biznesowego aplikacji goodie na komponenty techniczne chmury Azure. Celem współpracy było takie ich dobranie, aby skrócić czas udostępnienia aplikacji klientom oraz wykorzystać możliwie najnowsze, dostępne technologie"* – podkreśla Paweł Matulewicz, dyrektor ds. obsługi klientów w sektorze finansowym w polskim oddziale Microsoftu. Dzięki narzędziom Microsoftu możliwe stało się nie tylko oparcie goodie na bezpiecznej i skalowalnej infrastrukturze w chmurze, lecz także szybkie wdrożenie takich funkcji serwisu, jak: rozpoznawanie obrazu, silnik rekomendacji wykorzystujący narzędzia Machine Learning czy narzędzia do obsługi multimediów.

PRZYKŁADOWE NARZĘDZIA W AZURE ZASTOSOWANE W GOODIE

Wykorzystanie Azure Cognitive Services – Computer Vision API znacząco przyspieszyło np. proces digitalizacji obrazów z ulotek z supermarketów. Zebrane w ten sposób dane posłużyły następnie m.in. do porównywania cen. *„Gdybyśmy chcieli stworzyć podobne narzędzie we własnym zakresie, trwałoby to znacznie dłużej, niż korzystając od razu z gotowego komponentu"* – wyjaśnia Ricardo Campos. Z kolei Azure Cognitive Services – Recommendations API oraz Azure Machine Learning wykorzystano do stworzenia tzw. silników rekomendacji. Dzięki temu platforma goodie „uczy się" tego, co robią, lubią i jak zachowują się jej użytkownicy. *„Pozwala to nam optymalizować komunikaty, które od nas otrzymują, a to jedna z ważniejszych funkcjonalności platformy goodie"* – wyjaśnia Ricardo Campos.

OCENIĆ I PRZEANALIZOWAĆ PREFERENCJE KLIENTÓW

„Pokazujemy to, co dla użytkowników jest najważniejsze. Dzięki Azure Cognitive Services wiemy nie tylko to, że dana osoba lubi np. chińską

Bank Millennium skorzystał m.in. z takich narzędzi, jak:

- Azure **App Services – API Apps,**
- Azure **SQL Database,**
- Azure **DocumentDB** – NoSQL Database Services,
- Azure **Storage** (do przechowywania obrazów),
- Azure **Content Delivery Network,**
- Azure **Cognitive Services – Recommendations API,**
- Azure **Cognitive Services – Computer Vision API** (rozpoznawanie obrazów i przetwarzanie ich na tekst – OCR),
- Azure **Notification Hubs** (wysyłanie komunikatów typu push),
- Azure **Event Hub** (określanie lokalizacji użytkowników),
- Azure **SendGrid** (komunikacja mailowa z użytkownikami),
- Azure **Application Insights** (monitorowanie i analiza działania aplikacji goodie),
- Azure **Machine Learning** (wykorzystanie m.in. do przesyłania rekomendacji),
- Azure **Stream Services** (analiza danych w czasie rzeczywistym),
- Azure **Active Directory B2C** (identyfikacja użytkowników).

kuchnię. Nasza platforma – na bazie zachowań wszystkich użytkowników – potrafi znaleźć powiązania, nie wprost związane z kuchnią, a bardziej ze stylem życia danej grupy osób. Nie będziemy więc przesyłać informacji wyłącznie o promocjach w chińskich restauracjach w najbliższej okolicy. Chcemy, aby nasza platforma stała się częścią tzw. Customer Journey. Goodie ma być częścią tego procesu, częścią stylu życia naszych użytkowników i każdego z jego etapów" – wyjaśnia.

Już na początku korzystania z platformy goodie nowemu użytkownikowi pokazywane są obrazy konkretnych produktów i marek. W miarę jak są one oceniane, użytkownik otrzymuje kolejne zdjęcia. Wystarczy zaledwie kilka obrazów, aby wstępnie zanalizować jego preferencje. Poza tym goodie ma łączyć funkcjonalnie wiele dostępnych dziś aplikacji, ograniczonych zwykle do jednego miejsca (restauracji, sklepu) lub jednej marki. *„Dziś chcąc dokonać zakupu lub skorzystać z usługi, szukamy miejsca najbliżej nas, a następnie opinii na temat wybranego lokalu lub sklepu w mediach społecznościowych. Goodie ma być superaplikacją, łączącą wiele dostępnych niezależnie funkcji. Do tego – korzystając z rozwiązań typu Machine Learning lub Artificial Intelligence – można analizować i wykorzystywać dużo głębiej zbierane dane o preferencjach użytkowników"* – wyjaśnia Ricardo Campos.

KOMUNIKACJA Z GOODIE W JĘZYKU NATURALNYM

Obecny kształt platformy goodie to dopiero początek projektu. Planowany jest jej dalszy intensywny rozwój. W przyszłości goodie ma korzystać m.in. z chat botów. *„Zamiast szukać tego, czego potrzebujemy, wykorzystując menu tekstowe, będziemy mogli porozmawiać z goodie i – stosując zapytanie w języku naturalnym – powiedzieć np. chcę kupić spodnie na Mokotowie, goodie pomóż. Nasza platforma zaś pokaże nam najbliższe sklepy lub też zaproponuje zniżki na ulubione jeansy. Chcemy, aby goodie stało się częścią procesu podejmowania decyzji przez naszych użytkowników"* – mówi Ricardo Campos.

„Projektując i udoskonalając goodie, staramy się stworzyć nową jakość na rynku, m.in. dbając o wygodę użytkowania i jakość ofert dostępnych na platformie, a także tworząc nowe, niedostępne wcześniej rozwiązania i funkcjonalności. Chcemy, aby goodie stało się nieodłącznym towarzyszem podczas zakupów. Pomimo rozwoju e-handlu, Polacy nadal lubią kupować w galeriach handlowych, a dzięki goodie mogą do tego wykorzystywać nowe technologie, łącząc wygodę i dobre okazje" – podsumowuje Beata Krupińska, członek zarządu Millennium Goodie. Platforma goodie dostępna jest jako aplikacja na smartfony z systemami iOS i Android oraz jako nowoczesna platforma internetowa pod adresem www.goodie.pl.

Dzięki Azure Cognitive Services wiemy nie tylko to, że dana osoba lubi np. chińską kuchnię. Nasza platforma – na bazie zachowań wszystkich użytkowników – potrafi znaleźć powiązania, nie wprost związane z kuchnią, a bardziej ze stylem życia danej grupy osób. Nie będziemy więc przesyłać informacji wyłącznie o promocjach w chińskich restauracjach w najbliższej okolicy. **Chcemy, aby nasza platforma stała się częścią Customer Journey. Goodie ma być częścią tego procesu, częścią stylu życia naszych użytkowników i każdego z jego etapów.**

MACIEJ **LASYK**, DevOps Engineer w Ocado Technology
ADAM **JADCZAK**

KONTENERY

jako sposób na szybszą wirtualizację

Najbardziej popularnym rozwiązaniem do tworzenia kontenerów jest Docker. To lekkie środowisko wirtualizacyjne, stworzone z myślą o szybszym i prostszym wdrażaniu aplikacji i publikacji ich zmian. Bazuje na LXC, czyli kontenerach linuksowych. Microsoft i Docker podpisały porozumienie, aby wykorzystać zalety kontenerów Dockera w Windows Server 2016. Dzięki temu mechanizm Windows Server 2016 Containers wykorzystuje silnik Dockera, pozwalając na stosowanie go przez aplikacje stworzone dla systemu Windows.

Microsoft i Docker nawiązały współpracę zmierzającą do zapewnienia nowoczesnej platformy dla programistów i specjalistów IT. Pozwala ona na budowanie, udostępnianie i uruchamianie aplikacji rozproszonych zarówno w środowisku on-premise, jak i w chmurze. Ponadto dotyczy to systemu Windows i Linux. Wcześniej kontenery Dockera ograniczone były wyłącznie do tego drugiego systemu operacyjnego.

Partnerstwo obejmuje cały portfel narzędzi programistycznych Microsoft, systemów operacyjnych i usług w chmurze, w tym: Windows Server 2016, Hyper-V, Visual Studio i Microsoft Azure. Przykładowo – wraz z premierą Windows Server 2016 – w Azure Marketplace znalazły się: Docker Universal Control Plan (UCP), Docker Trusted Registry (DTR) i Docker Engine. Dzięki temu aplikacje dekstopowe można przenieść do chmury Azure bez konieczności dokonywania jakichkolwiek zmian w ich kodzie. Dzięki zaś aplikacji Docker for Azure można wykorzystać platformę Docker do uruchamiania aplikacji w infrastrukturze IaaS Microsoft Azure.

Współpraca Microsoft i Dockera to także korzyści dla programistów. Użytkownicy Dockera moga zastosować elementy tego samego klienta (CLI) do budowy aplikacji zarówno na potrzeby systemów Windows, jak i Linux. Przypomnijmy, Docker Engine to aplikacja typu klient-serwer, złożona z trzech elementów: serwera (daemona), REST API i właśnie klienta (CLI). Klient Dockera i graficzny interfejst Kitematic GUI wspierane są bowiem obecnie w środowisku Windows. Jednocześnie programiści Microsoftu mogą wykorzystać swoje ulubione narzędzia programistyczne – jak Visual Studio – do budowy aplikacji uruchamianych w kontenerach Dockera.

W kontenerach dockerowych uruchomić można w prosty sposób praktycznie dowolną aplikację. Ponadto metoda budowania obrazów została tak przemyślana, aby były one cache'owane w trakcie budowania. Dzięki temu kolejne procesy tworzenia obrazów przebiegają bardzo szybko. Mogą bowiem bazować na poprzednikach. Systemy plików poszczególnych kontenerów są oddzielne, w związku z tym mamy wrażenie pracy na zupełnie oddzielnym systemie operacyjnym.

CZYM JEST WIRTUALIZACJA OPARTA NA KONTENERACH?

Wirtualizacja oparta na kontenerach używa jądra systemu operacyjnego hosta do uruchamiania wielu instancji aplikacji gości. Główną zaletą Dockera jest fakt, że stworzone za jego pomocą środowiska wirtualne będą działały i zachowy-

wały się dokładnie tak samo, niezależnie od tego, w jakim środowisku zostaną uruchomione. Kontenery dockerowe można uruchomić zarówno na maszynach dedykowanych, jak i w dowolnego rodzaju maszynach wirtualnych. We wszystkich tych miejscach ich zachowanie będzie niezmienne. Dzięki temu wystarczy raz zbudować środowisko dla wybranej aplikacji – potem można już tylko korzystać z tak przygotowanego szablonu. Migracja kontenerów dockerowych jest bardzo prosta.

Kolejną zaletą jest wielozadaniowość. W kontenerze można uruchomić wiele popularnych dystrybucji Linuksa (a ostatnio także system Windows dzięki Windows Server 2016 – przyp. red.), jednak muszą one być odpowiednio przygotowane. Jeśli nie mamy czasu na pracę nad własną modyfikacją, istnieje pokaźna biblioteka gotowych obrazów i szablonów. Pozwala to w kontenerach dockerowych uruchomić w prosty sposób praktycznie dowolną aplikację. Ponadto metoda budowania obrazów została tak przemyślana, aby były one cache'owane w trakcie budowania. Dzięki temu kolejne procesy tworzenia obrazów przebiegają bardzo szybko. Mogą bowiem bazować na poprzednikach. Systemy plików poszczególnych kontenerów są oddzielne, w związku z tym mamy wrażenie pracy na zupełnie oddzielnym systemie operacyjnym.

KONTENERY OPARTE NA DOCKERZE VS LXC

Po co w ogóle Docker, skoro bazuje on na LXC i praktycznie nie rozwija jego funkcjonalności? Odpowiedź jest prosta – Docker zmienia kompletnie podejście do tworzenia i utrzymania środowiska wirtualnego.

1. Docker jest warstwą abstrakcji, która mocno upraszcza i ułatwia zarządzanie kontenerami. Przykładowo, uruchamiając kontener dockerowy na różnych środowiskach – zachowa się on zawsze tak samo. W przypadku LXC nie mamy tej pewności, ponieważ środowiska oparte na LXC mogą się znacząco różnić.
2. System wersjonowania kontenerów dockerowych bardzo ułatwia prace zarówno programistom, jak i innym osobom zajmującym się utrzymaniem środowisk. Dotyczy to tworzenia nowych wersji, commitowania czy przywracania starych.
3. Istnieje możliwość wielokrotnego użycia w projektach podstawowych obrazów. Wystarczy stworzyć pewien szablon, na bazie którego można opracować późniejsze gałęzie innych wersji oprogramowania.

Docker – jak widać – daje sporo elastyczności obudowanej w łatwy do użycia interfejs. Większość funkcji można zasymulować za pomocą czystego LXC, ale to kosztuje sporo pracy, którą już w przypadku Dockera wykonano za nas. Używanie Dockera dla programistów bywa dużo przyjemniejsze i obarczone mniejszym ryzykiem zawału czy nerwicy.

DLA KOGO DOCKER I JAKIE MA ZASTOSOWANIA?

Docker pozwala stworzyć bardzo zaawansowane środowiska, składające się z wielkiej liczby kontenerów hostujących skomplikowane i ciężkie aplikacje. Co więcej – za pomocą tych kontenerów można z łatwością tworzyć środowiska deweloperskie i testowe.
Wyobraźmy sobie scenariusz, w którym zespół SCRUM pracuje nad aplikacją. Środowisko dla tej aplikacji można z łatwością zbudować w Dockerze, a następnie rozesłać między członkami zespołu. Można dokonać tego czy to w postaci pliku tar, czy też używając prywatnego repozytorium, które to Docker wspiera. Gdy aplikacja jest gotowa, programiści wysyłają zmiany do systemu kontroli wersji, a ten komunikuje się z systemem do walidowania poprawności kodu i testowania wprowadzonych zmian oraz próbuje wykonać budowę i testy aplikacji. Takim systemem może być np. Jenkins, i może on w łatwy sposób po swojej stronie budować kontenery do testowania aplikacji.

Sam Jenkins również może być uruchomiony w kontenerze. Docker wspiera zagnieżdżoną konteneryzację. Takie środowiska testowe buduje się bardzo szybko, a testowanie jest łatwiejsze, ponieważ wirtualizacja nie ma tak sporego narzutu. Na koniec – publikacja nowej wersji aplikacji w środowisku produkcyjnym również przebiega błyskawicznie, gdyż Docker z repozytorium pobierze tylko te zmiany, które zostały wprowadzone w aplikacji, i dopisze je do ostatniej wersji, tworząc nowy obraz. Proces jest wyjątkowo szybki.

Programista może więc bez problemu poradzić sobie z procesami CI/CD oraz zarządzać wersjonowaniem w środowisku produkcyjnym, pozostawiając administratorom jedynie konfigurowanie i poprawę wydajności systemu hosta.

JAK STWORZYĆ SWÓJ PIERWSZY KONTENER DOCKERA

Zarządzanie kontenerami jest bardzo proste – interfejs CLI ma dość bogatą dokumentację, ale liczba funkcji nie jest przygniatająca. Dzięki temu łatwo o szybki start dla osób, które z Dockerem nigdy nie miały do czynienia. Poniżej opis, jak zbudować prosty kontener. Najpierw należy utworzyć plik o nazwie Dockerfile i opisać w nim, w jaki sposób kontener ma zostać zbudowany. Oto jego kod:

```
`FROM fedora
MAINTAINER scollier <scollier@redhat.com>

RUN yum – y update; yum clean all
RUN yum – y install nginx; yum clean all
RUN echo „daemon off;" >> /etc/nginx/nginx.conf
RUN echo „nginx on Fedora" > /usr/share/nginx/html/index.html

EXPOSE 80

CMD [ „/usr/sbin/nginx" ]`
```

W ten oto sposób stworzono konfigurację pierwszego kontenera, którego jedynym zadaniem jest uruchomienie serwera Nginx i serwowanie statycznej strony index.html o treści „nginx on Fedora". Teraz należy zbudować środowisko oparte na tym Dockerfile, a następnie je uruchomić.

```
`$ docker build – – rm – t nginx .
$ docker run – d – p 80:80 – i nginx`
```

W ten oto sposób utworzono i uruchomiono pierwszy w pełni funkcjonalny kontener, którego jedyną funkcją jest serwowanie statycznej strony za pomocą serwera Nginx. Co do szczegółów dotyczących Dockera, większość akcji związanych z zarządzaniem kontenerami wykonujemy za pomocą binarki ‚docker'. Ma ona wiele parametrów, które, oczywiście, można odnaleźć w dokumentacji. Najważniejsze z nich to:

build – służy do tworzenia obrazu kontenera z utworzonego Dockerfile,
run – pozwala na uruchomienie instancji kontenera z przygotowanego obrazu,
attach – komenda ta umożliwia podłączenie się do uruchomionej instancji kontenera,
commit – utworzona instancja kontenera po wyłączeniu nie zapisze żadnych zmian w swoim wewnętrznym filesystemie – zostaną one utracone (taka jest idea); dlatego też, jeśli w trakcie pracy kontenera zechcemy zapisać jego stan, to za pomocą tej komendy możemy utworzyć nowy obraz kontenera z wybranej instancji,
export – pozwala wyeksportować kontener do pliku tar,
history – istotna komenda, dzięki której możemy zobaczyć historię danej instancji kontenera,
images – listowanie utworzonych na serwerze obrazów,
import – importowanie kontenerów (np. z plików tar, tar.gz, tgz, bzip itd.),
ps – wyświetlanie istniejących instancji kontenerów.

Dostępnych komend jest oczywiście dużo więcej, dlatego zachęcam do przeglądu dokumentacji.

AUTOMATYZACJA PROCESÓW DZIĘKI KONTENEROM

Obecnie standardem przemysłowym jest automatyzowanie procesów. W środowiskach IT mamy do tego wiele znanych narzędzi: Chef, Puppet, Cfengine czy popularny ostatnio Ansible. Ansible – jako narzędzie wśród wymienionych – najlżejsze i również obarczone najłatwiejszą ścieżką nauki. Idealnie wpisuje się ono w świat kontenerów dockerowych. Zawiera ponadto gotowe moduły do obsługi środowisk opartych na Dockerze, więc ilość pracy, którą trzeba wykonać – aby wdrożyć automatyzację – drastycznie maleje.

CO KRYJE ŚRODEK DOCKERFILE?

Dockerfile jest plikiem zawierającym instrukcje wymagane do utworzenia obrazu kontenera. Na powyższym przykładzie można zauważyć pięć różnych instrukcji.

FROM fedora – oznacza, aby z publicznego repozytorium obrazów pobrać dystrybucję Fedora (ale może być to inny system operacyjny), której będziemy używać jako system tego kontenera.
MAINTAINER – informacja na temat osoby lub grupy opiekującej się danym kontenerem.
RUN – uruchomienie wybranej komendy systemowej już w nowym systemie kontenera.
EXPOSE – prosty sposób na konfigurację nasłuchiwania na wybranych portach.

CMD – określa aplikację uruchomieniową, czyli tę, która ma działać na serwerze.

Ideą Dockera jest uruchamianie jednej aplikacji na kontenerze, stąd tylko ostatnie polecenie CMD zostanie wykonane. Aby uruchomić więcej niż jedną aplikację, należy w CMD użyć np. demona supervisord. Takich poleceń wewnętrznych Dockerfile jest dużo więcej. Dzięki nim możemy definiować współdzielone z innymi kontenerami (lub hostem) katalogi i zmienne środowiskowe dla kontenera, czy przegrywać w trakcie procesu tworzenia pliki z hosta na kontener. Składnia jest elastyczna, dzięki czemu łatwo jest utworzyć środowisko dla znanych aplikacji.

ZARZĄDZANIE ZASOBAMI KONTENERÓW

W LXC do zarządzania zasobami używamy control – groups. Podobnie ma to miejsce w Dockerze, jednak jest to tutaj trochę prostsze. Silnikiem do dystrybuowania zasobów jest ciągle cgroups. Aby ustawić limit pamięci dla uruchamianej instancji kontenera, wystarczy podać parametr uruchomieniowy, np. `docker run – m="512m"`. Możliwe jest też zarządzanie zasobami, takimi jak priorytet procesora.

BEZPIECZEŃSTWO W DOCKERZE

Docker bazuje na kilku rozwiązaniach zapewniających bezpieczeństwo. Pierwszym z tych rozwiązań są linuksowe przestrzenie nazw (namespaces). Zapewniają one izolację pomiędzy kontenerem a całym środowiskiem zewnętrznym (a więc i kontenerami stojącymi obok). Dzięki temu właśnie kontenery mają wrażenie, że zawierają zupełnie normalne (choć trochę ograniczone) zasoby systemowe, gdzie w rzeczywistości są to tylko specjalnie spreparowane widoki (np. na system plików czy sieć).

Kolejną furtką, która chroni kontenery, jest sposób zarządzania zasobami. Został on w całości oparty na linuksowych control – groups, a więc też przebywa na barkach LXC. Dzięki użyciu cgroups kontenerom można ograniczać zasoby, takie jak pamięć, cykle procesora, sieć, czy nawet operacje typu I/O. Te możliwości są o tyle istotne, że rezerwacja zasobów dla wybranych kontenerów pozwala im np. zapewnić wysoką stabilność i wykonanie SLA w części dotyczącej zasobów.

Co istotne, każda technologia wirtualizacji czy konteneryzacji ma słabe strony i naszym zadaniem – jako operatorów – jest zapewnienie, aby jeszcze przed wdrożeniem żadne z nich nie ujrzały światła dziennego. W przypadku LXC czy Dockera rozsądnie jest np. dodać jeszcze jakąś warstwę abstrakcji, która dodatkowo zaizoluje już mocno zapewnioną izolację kontenerów – np. za pomocą Linux Security Modules (LSM) typu SELinux czy GrSec/PAX lub AppArmor. Największym zagrożeniem wciąż pozostaje błąd ludzki lub brak doświadczenia i wiedzy na temat uruchamianego środowiska – i tutaj najłatwiej jest szukać pomysłów na ekslpoatację systemów.

ZARZĄDZANIE SIECIĄ W KONTENERACH

Docker używa mostków sieciowych (network bridge) do zapewnienia komunikacji. Domyślnie tworzonym mostkiem jest docker0. Przed jego utworzeniem Docker szuka wolnej klasy adresowej niepozostającej w konflikcie z obecnymi (na podstawie aktualnej tablicy routingu). Następnie podejmuje decyzje o jej zajęciu i binduje się do jednego z adresów IP w tej klasie (np. 172.17.12.0/24).

Wyobraźmy sobie scenariusz, w którym zespół SCRUM pracuje nad aplikacją. Środowisko dla tej aplikacji można z łatwością zbudować w Dockerze, a następnie rozesłać między członkami zespołu. Można dokonać tego czy to w postaci pliku tar, czy też używając prywatnego repozytorium, które to Docker wspiera. **Gdy aplikacja jest gotowa, programiści wysyłają zmiany do systemu kontroli wersji, a ten komunikuje się z systemem do walidowania poprawności kodu i testowania wprowadzonych zmian** oraz próbuje wykonać budowę i testy aplikacji.

Dla każdej nowej instancji kontenera uruchamiany jest nowy wirtualny interfejs sieciowy, który jest bondowany do głównego mostka. Jego adres (docker0) jest wykorzystany jako brama. Ponadto Docker używa iptables do zarządzania komunikacją między portami, wybranymi kontenerami itd.

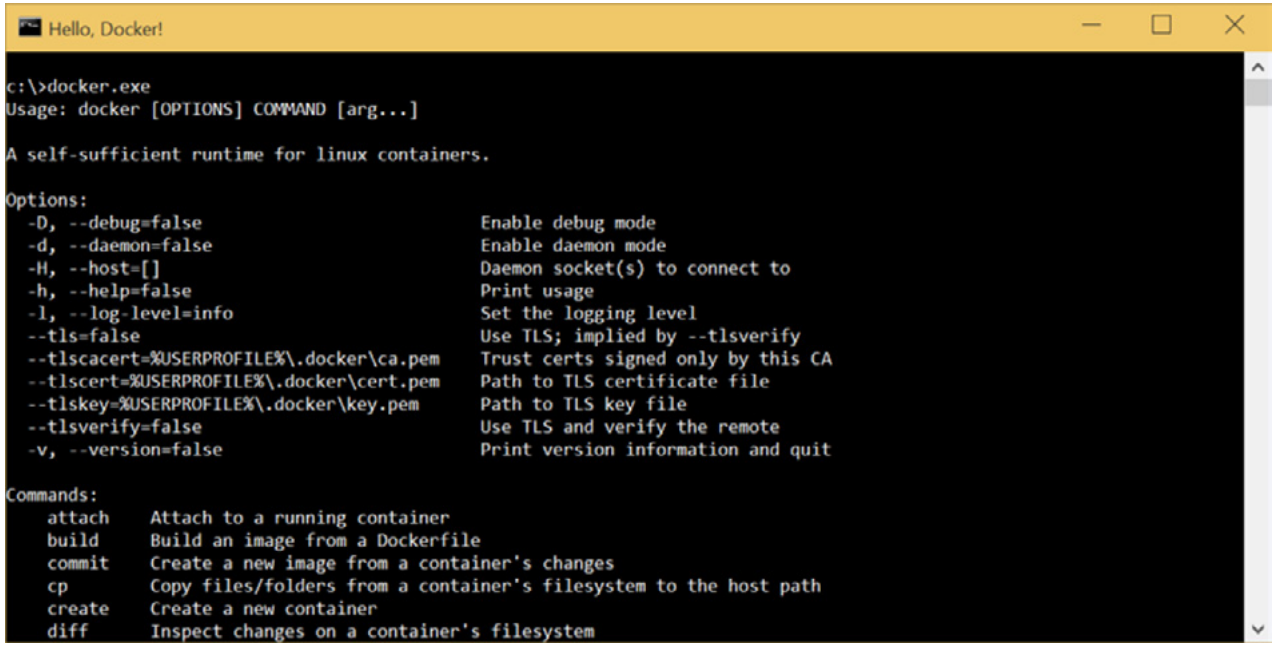
W ten sposób zapewniona jest możliwość komunikacji między kontenerami i tej przychodzącej z zewnątrz. Oczywiście, istnieje też możliwość modyfikowania ustawień sieciowych (np. zmiana zakresu sieci). Podobnie w ostatniej wersji 0.11 (która jest RC) dodano tryb host networking. Dzięki temu kontener może działać w ramach sieci hosta, a nie samego Dockera. To może ułatwić konfigurację środowiska.

REPOZYTORIUM OBRAZÓW

Obrazy kontenerów można łatwo eksportować i importować (za pośrednictwem komend import/export). Można również użyć do tego centralnego repozytorium, do którego publikujemy swoją pracę. Docker zawiera publiczne repozytorium obrazów pod adresem index.docker.io – znajdziemy w nim bardzo dużą liczbę obrazów, które zostały opublikowane przez społeczność. Istnieje też możliwość zamówienia opcji prywatnego repozytorium.

Często jednak interesuje nas utworzenie prywatnego repozytorium w ramach własnej infrastruktury IT. Jest również taka możliwość. Korzysta ona z projektu ‚docker-registry', który znajduje się na GitHubie pod oficjalnym kontem firmy DotCloud – właściciela Dockera. Ponadto w internecie można znaleźć kilka metod na utworzenie prywatnych repozytoriów, np. za pomocą już gotowych kontenerów.

PODSTAWOWE KOMENDY DOCKERA



Moje dokonania są widoczne m.in. w projekcie Fedora-Dockerfiles dostępnym na GitHubie: https://github.com/fedora-cloud/Fedora-Dockerfiles. Jest to doskonałe miejsce do rozpoczęcia prac z Dockerem, gdyż zawiera bogatą liczbę przykładów, w jaki sposób można zbudować i uruchomić proste kontenery. Przykład opisany w artykule został wzięty właśnie z tego projektu. Oficjalna dokumentacja dostępna jest na docs.docker.io.

ZALETY ZASTOSOWANIA KONTENERÓW

- są lżejsze niż maszyny wirtualne,
- nie wymagają instalacji systemu operacyjnego (Guest OS),
- mają mniejsze wymagania względem zasobów (CPU/RAM),
- umożliwiają większą gęstość ich upakowania na pojedynczym hoście,
- pozwalają na dużo łatwiejszą ich przenaszalność.

PAWEŁ **OKOPIEŃ**

RIGHTSCALE:

chmury hybrydowa i publiczna coraz popularniejsze w roku 2017

RightScale opublikował najnowszy raport State Of The Cloud 2017 na temat wykorzystania chmury przez przedsiębiorstwa i korporacje. Wynika z nich, że coraz większą popularnością cieszy się chmura publiczna. Firmy odchodzą zaś powoli od rozwiązań prywatnych. Beneficjentem zmian w podejściu do IT jest Microsoft i usługa Azure, której popularność w dużych firmach wzrosła w ostatnim roku z 26% do 43%.

RightScale 2017 State of The Cloud to raport będący podsumowaniem badania na ponad 1000 specjalistów IT zarówno z dużych korporacji (ponad 1000 pracowników), jak i małych, i średnich firm. Respondenci odpowiadali na pytania związane z: wykorzystywanym środowiskiem IT, planami odnośnie do użycia usług cloud computing, motywów decyzji dotyczących wykorzystania chmury i tego, jak są one postrzegane.

SPADEK POPULARNOŚCI CHMURY PRYWATNEJ

Jedną z najważniejszych informacji, jaką dostarcza raport RightScale State Of The Cloud 2017, jest rodzaj chmury, z jakiej korzystają przedsiębiorstwa. Tylko w 5% jest to wyłącznie chmura prywatna, w przeważającej większości użytkowana jest chmura hybrydowa (67%, 71% przed rokiem), a 22% korzysta wyłącznie z chmury publicznej. Łącznie odsetek osób korzystających z chmury prywatnej zmniejszył się – w stosunku do roku 2016 – z 77% do 72%. Z chmury publicznej korzysta zaś 89% ankietowanych organizacji (także 89% w roku 2016). Zgodnie z raportem State Of The Cloud 2017, większość obciążeń (41%) odbywa się w chmurze publicznej, a dopiero na drugim miejscu jest chmura prywatna (38%).

Biorące udział w badaniu RightScale firmy w zdecydowanej większości nie korzystają z jednej chmury, a z wielu tego typu rozwiązań (85%), w tym 58% stanowią rozwiązania hybrydowe. Tylko 9% ankietowanych używa wyłącznie jednego rozwiązania cloud computing opartego o chmurę publiczną, a 5% o chmurę prywatną. Z raportu State Of The Cloud 2017 wyraźnie widać, że chmura prywatna jest w odwrocie. Bardzo mało przedsiębiorstw rozważa także budowanie własnych rozwiązań typu private cloud (9%). Aż 50% ankietowanych przez RightScale będzie inwestować w roku 2017 w chmury hybrydowe!

95%

organizacji ankietowanych w badaniu RightScale State Of The Cloud 2017 wykorzystuje lub eksperymentuje z aplikacjami działającymi na platformach oferowanych w modelu Infrastructure-as-a-Service.

50%

działów IT – wg raportu State Of The Cloud 2017 – planuje wdrożenie rozwiązań chmury hybrydowej, a tylko 9% prywatnej.

67%

to wg raportu RightScale udział chmury hybrydowej w dużych i średnich firmach (71% rok wcześniej). Hybrid cloud jest preferowaną przez duże firmy strategią podejścia do chmury. Równocześnie udział chmury prywatnej spadł z 77% w roku 2016 do 72% na początku 2017 r.

WYZWANIEM ZARZĄDZANIE KOSZTAMI CHMURY, BEZPIECZEŃSTWO I BRAK ZASOBÓW

Badani wciąż dostrzegają korzyści płynące z użytkowania chmury. Opinie na ten temat niemal nie zmieniają się na przestrzeni ostatnich lat. Nadal najważniejsze atuty to szybszy dostęp do infrastruktury, lepsza skalowalność, wyższa dostępność. Jednocześnie korzystanie w pełni z usług cloud computing pozwala zweryfikować pewne opinie. Coraz mniej respondentów uważa, że chmura pozwala zredukować koszty. W 2017 roku taką opinię miało 35% ankietowanych, w 2016 roku 37%. Podobnie rozkładają się odpowiedzi, jeśli chodzi o efektywność działów IT. Zalety te szczególnie odczuwają firmy, które już intensywnie funkcjonują w chmurze.

Ciekawe jest to, że chmura przestaje się kojarzyć z zagrożeniem bezpieczeństwa. Kwestia bezpieczeństwa jest ważna przede wszystkim dla firm, które dopiero planują wdrożenie rozwiązania cloud computing (32%). Nie obawiają się o bezpieczeństwo tak mocno firmy skupione na chmurze. Uważa tak jedynie 19% respondentów i odpowiedź ta jest dopiero na 4. pozycji po zarządzaniu kosztami chmury (24%), zachowaniu compliance (22%) i braku niezbędnych zasobów i specjalistów (21%). Jeszcze w 2015 roku kwestie bezpieczeństwa jako znaczące wyzwanie dla rozwiązań w chmurze uznawało 41% badanych. W 2017 roku jest to już tylko 35% respondentów.

Dla firm operujących już w pełni w chmurze (Cloud Focused) kluczowe jest zarządzanie kosztami. Kwestia ta pojawia się jednak także wśród priorytetów u pozostałych respondentów. W przypadku Cloud Explorers jest na 2. pozycji (27%), a w kategorii Cloud Beginners na 3. (25%). Zdaniem ankietowanych przez RightScale, aż 30% zasobów udostępnianych w chmurze marnuje się. Według analityków RightScale może to być nawet 45%. W celu optymalizacji kosztów wykorzystania chmury firmy biorące udział w badaniu planują: rezygnację z niewykorzystywanych zasobów albo przeniesienie zasobów do dostawców oferujących tańsze rozwiązania lub w krajach, gdzie te usługi są tańsze.

Dla firm, które już korzystają z chmury, ale wciąż rozwijają się w tym obszarze (Cloud Explorers) najważniejszym wyzwaniem jest brak odpowiednich zasobów (28%). We wszystkich ankietach pojawia się także kwestia możliwości kontroli i zarządzania chmurą.

DOCKER KLUCZOWYM NARZĘDZIEM DEVOPS

Zdecydowana większość przedsiębiorstw w kontekście chmury korzysta w swojej pracy z metodologii DevOps. Odsetek firm wybierających ten styl wzrósł w ciągu roku z 74% do 78%. Jeszcze lepiej sytuacja wygląda w dużych firmach, gdzie adaptacja DevOps wynosi aż 84%! Z tego w 30% dużych firm metodologia ta obejmuje całą fir-

65%

przedstawicieli działów IT ankietowanych przez RightScale uważa, że ich rola w zakresie rozwiązań cloud computing dotyczyć będzie wyboru chmury publicznej (55% rok wcześniej), a w 63% przypadków - doradzania w sprawie przenoszenia aplikacji do chmury (55%).

mę, w 28% poszczególne działy, a w 27% tylko konkretne zespoły lub projekty.

Z badania przeprowadzonego przez RightScale wynika też, że kluczowym narzędziem pozwalającym na standaryzację i automatyzację wdrażania oraz konfiguracji serwerów i aplikacji jest Docker. Stosuje go 35% respondentów, a kolejne 32% zamierza wdrożyć. Na drugim miejscu jest Chef (odpowiednio 28% i 17%), na trzecim Puppet (28%, 17%). W porównaniu z rokiem ubiegłym adaptacja Dockera wzrosła z 27% do 35%. Wzrost zanotowało też wykorzystanie Kubernetes (z 7% do 14% ankietowanych). Użytkowanie pozostałych popularnych narzędzi pozostało na tym samym poziomie lub spadło.

Znacznie szersze wykorzystanie Dockera widać w korporacjach. Używa go 40% dużych firm, a kolejne 40% ma zamiar zacząć. W tym segmencie respondentów Docker również notuje duże wzrosty rok do roku (z 29% do 40%). Znacząco za to zmalało wykorzystanie narzędzia Puppet (z 42% do 37%). Docker króluje jednak niezależnie od wielkości przedsiębiorstwa.

MICROSOFT AZURE ROŚNIE W SIŁĘ

Coraz popularniejsze staje się też stosowanie – w kontekście kontenerów – usług typu Container-as-a-Service. Sporo firm również zamierza rozpocząć takie użytkowanie. Z AWS ECS korzysta 35% ankietowanych (27% planuje). Z kolei z Azure Container Service 11% (28% planuje). Po Google GKE sięga zaś 8% (kolejne 18% planuje). Szersze zainteresowanie Azure odzwierciedla coraz większą popularność platformy Microsoftu.

W korporacjach Microsoft Azure zwiększyło udziały w chmurze publicznej z 26% do 43%. W tych największych firmach także adaptacja Amazon Web Services wzrosła z 56% do 59%. Jednak gdy przyjrzymy się wszystkim przedsiębiorstwom, to Azure zostało wdrożone w 34% firm (rok temu jedynie 20%). Tymczasem udział w rynku AWS zatrzymał się na 57%. Wyraźnie rośnie też zainteresowanie Google Cloud (z 10% do 15%).

CHMURA MA SIĘ ŚWIETNIE I JEJ POPULARNOŚĆ NABIERA ZNACZENIA

Jak wynika z badania RightScale, usługi cloud computing na dobre zadomowiły się w firmach. Tylko 6% przedsiębiorstw nie ma planów związanych z chmurą, 14% ankietowanych dopiero przygląda się chmurze i planuje wykorzystanie jej w przedsiębiorstwie, jednak aż 33% to firmy z kategorii Cloud Focused, a 25% firm aktywnie korzysta z chmury (Cloud Explorers), natomiast 22% jest dopiero na początku swej przygody z cloud computing. Coraz większa ma być też w roku 2017 rola działów IT w zakresie rozwiązań cloud computing. Zdaniem aż 65% przedstawicieli IT, dotyczyć będzie ona wyboru chmury publicznej (55% rok wcześniej), a w 63% przypadków – doradzania w sprawie przenoszenia aplikacji do chmury (55%).

35%

ankietowanych przez RightScale stosuje jako narzędzie do tworzenia kontenerów Dockera. To więcej niż narzędzia Chef i Puppet (po 28% każdy, rok temu było to 32%).

Wśród respondentów RightScale dominowały osoby z dużych przedsiębiorstw zatrudniających powyżej 1000 osób (48%), drugą grupę stanowiły małe przedsiębiorstwa do 100 pracowników (32%), najmniej ankietowanych zajmowało się infrastrukturą IT w firmach zatrudniających od 100-1000 osób (21%). Badano organizacje z różnych sektorów. Przepytano reprezentantów z firm głównie stacjonujących w Ameryce Północnej (61%), ale w badaniu przyjrzano się również firmom z Europy, Azji i innych części świata. Wśród ankietowanych znaleźli się szefowie działów IT, programiści, architekci systemów i inni pracownicy związani z IT. Badanie przeprowadzono w styczniu 2017 roku.

MARCIN **MARCINIAK**

NANO SERVER W WS 2016:

mały, chmurowy i zdalnie zarządzany

Microsoft kładzie coraz większy nacisk na usługi chmurowe, zwłaszcza w kontekście chmury hybrydowej łączącej własne zasoby ze środowiskiem cloud computing. Wprowadzone niedawno wydanie serwera Nano doskonale wpisuje się w ten trend. Nano Server to nowy sposób instalacji edycji Standard i Datacenter Windows Server 2016. Został zbudowany specjalnie pod kątem chmury i budowy kontenerów w środowisku Windows współpracujących z Dockerem.

Wraz z premierą Windows Server 2016 w październiku 2016 roku wprowadzono możliwość instalacji tego serwera w wersji Nano. Jest to najmniejsza jego edycja ze wszystkich dostępnych. Waży zaledwie kilkaset MB. Wydanie Server Core – obecne w poprzednich wersjach systemów serwerowych Microsoftu – charakteryzowało się zminimalizowaną instalacją, pozbawioną graficznego interfejsu użytkownika. Składniki te można było jednak doinstalować, o ile taka była potrzeba. Wydanie Nano nie ma tej możliwości, ale w zamian za to charakteryzuje się znacznie mniejszym tzw. narzutem, czyli poziomem użycia pamięci, CPU i miejsca na dysku spowodowanym obecnością tych dodatkowych składników systemu. Minimalizacja narzutu przekłada się wprost na mniejsze zużycie zasobów i lepszą sprawność całego środowiska.

Z wydania Nano Microsoft usunął nie tylko interfejs użytkownika i lokalną konsolę. Windows Server 2016 zainstalowany w wersji Nano nie ma także warstwy kompatybilności z aplikacjami 32-bitowymi. Nie obsługuje również instalacji za pomocą narzędzia MSI (Windows Installer). Powstała w ten sposób niezwykle ograniczona wersja systemu. Bardzo dobrze pasuje ona więc do środowisk o wysokiej gęstości wirtualizacji. W środowisku Windows Server 2016 będzie możliwe uruchamianie nawet kilkuset maszyn wirtualnych – zainstalowanych jako Nano Server – na pojedynczym hoście.

LEPSZE BEZPIECZEŃSTWO, MNIEJ RESTARTÓW SYSTEMU

Usunięcie zbędnych – dla takiego zastosowania usług – nie pozostaje bez wpływu na poprawę bezpieczeństwa całego środowiska. Wiele z usług, które musiały pracować w standardowym wydaniu serwera Windows, w trybie instalacji Nano jest zbędnych. Nie trzeba będzie zatem wprowadzać ich aktualizacji. Zmiana ta jest bardzo istotna z punktu bezpieczeństwa właśnie. Według Microsoftu, aż 93% poprawek krytycznych dla bezpieczeństwa nie będzie bowiem obejmować tego wydania. Mniejsza liczba aktualizacji przekłada się także na mniejszą – aż o 80% – liczbę koniecznych restartów.

TYPOWE ZASTOSOWANIA NANO SERVERA

Nano Server zaprojektowany został pod kątem wysokiej gęstości wirtualizacji (kilkaset maszyn na jednym hoście), automatyzacji i szybkości wdrożenia (czas instalacji skrócono z ok. 19 minut do ok. 40 sekund). Ograniczenie dostępnych składników systemu przekłada się jednak na zastosowanie tej najmniejszej edycji serwera Windows. Obecnie może ona pełnić rolę natywnego serwera aplikacyjnego w klastrowym środowisku wirtualizacji Hyper-V oraz serwera plików i drukarek.

ZDALNE ZARZĄDZANIE MASZYNAMI WIRTUALNYMI

Usunięcie interfejsu użytkownika oznacza, że taką instalacją trzeba zarządzać zdalnie za po-

mocą odpowiednich narzędzi, także pracujących w środowisku graficznym. Jednym z nich są standardowe przystawki do konsoli MMC (Microsoft Management Console), takie jak menedżer usług, Desired State Configuration, zarządzanie serwerem, a także nieodłączna konsola PowerShell i system zarządzania Microsoft System Center.

Wydaniem Nano można też zarządzać za pomocą narzędzi firm trzecich, takich jak Puppet i Chef. Zarządzanie lokalne polega jedynie na bardzo ograniczonej konsoli Nano Server Recovery Console, która obsługuje jedynie podstawową – niezbędną do uruchomienia zdalnego zarządzania – funkcjonalność. Do zdalnego zarządzania administratorzy najczęściej korzystają albo z wiersza poleceń PowerShell, albo popularnej i znanej konsoli MMC z odpowiednimi przystawkami.

NANO SERVER MOŻNA JUŻ TESTOWAĆ

Aby zainstalować Windows Server 2016 w wersji Nano, należy wydobyć katalog Nano Server z obrazu ISO płyty instalacyjnej systemu Windows Server 2016. Następnie za pomocą skryptu PowerShell trzeba utworzyć obraz instalacyjny.

Przykład stworzenia obrazu ISO Nano Servera przy użyciu narzędzi Vagrant

```
Import-Module NanoServerImageGenerator.psd1
$adminPassword = ConvertTo-SecureString „vagrant” – AsPlainText – Force

New-NanoServerImage `
 – MediaPath D:\ `
 – BasePath .\Base `
 – TargetPath .\Nano\Nano.vhdx `
 – ComputerName Nano `
 – Package @(‚Microsoft-NanoServer-DSC--Package’,’Microsoft-NanoServer-IIS-Package’) `
 – Containers `
 – DeploymentType Guest `
 – Edition Standard `
 – AdministratorPassword $adminPassword
```

Dodatkowo w tym samym obrazie Nano Server dostępne są dodatkowe paczki instalacyjne, które będzie można dodać do instancji Nano za pomocą dyrektywy Add-WindowsPackage. Proces instalacji instancji Nano trwa – na odpowiednio szybkiej maszynie – ok. 40 sekund, co jest znaczącym przyspieszeniem w porównaniu do wersji z interfejsem użytkownika. Czas instalacji w tym przypadku to ok. 19 minut.

Nano Server szczególnie dobrze sprawdzi się w zastosowaniach takich jak:

- węzeł obliczeniowy hostowany w środowisku wirtualizacji Hyper-V (zarówno w konfiguracji klastrowej, jak i bez niej),
- węzeł storage dla usługi Scale-Out File Server,
- serwer DNS,
- serwer webowy IIS,
- host dla aplikacji chmurowych uruchamianych w kontenerze lub w maszynie wirtualnej gościa.

Wydanie Nano może też pracować na fizycznym komputerze, ale proces instalacji będzie nieco bardziej skomplikowany z racji na konieczność dostarczenia dodatkowych sterowników. Proces ten nie jest szczególnie skomplikowany, ale wymaga odpowiedniego przygotowania obrazu pod kątem docelowej maszyny. Podobnie proces dołączania do domeny jest nieco bardziej skomplikowany niż w przypadku normalnej instalacji Windows, ale taka konfiguracja również jest możliwa i wspierana.