Bygging, utrulling og provisjonering

Fagdag BEKK 08.11.2013

Dokumentet inneholder Kom-i-gang instruksjoner og oppgaver for heldagsworkshop rundt Bygging, utrulling og provisjonering i .NET.

Nøkkelord: Team City, Octopus Deploy, Powershell, Visual Studio

Krav:

- Visual Studio 2012 eller nyere
- Git, f.eks en av følgende:
    - Git for Windows (f.eks http://windows.github.com/)
    - Git (http://git-scm.com/download/win)

BEKK CONSULTING AS
P.O. BOX 134 SENTRUM, 0102 OSLO, NORWAY. PHONE +47 23 35 77 00, FAX +47 23 35 77 01, post@bekk.no, www.bekk.no

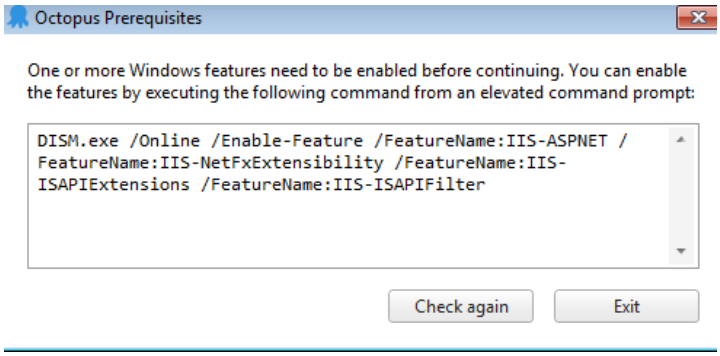
Kom-i-gang (Hyper-V image – kun Windows 8)

1. Last ned image og pakk ut zip-filen
2. Kjør kompatibilitetssjekk
    a. Åpne command prompt i windows.
    b. Skriv inn systeminfo.exe, trykk enter.
    c. Hvis Hyper-V allerede er installert vises *A hypervisor has been detected.* Du kan da gå videre til punkt 3.
    d. Dersom Hyper-V ikke er installert vil du se fire krav til Hyper-V. For å kunne kjøre imaget må alle de fire punktene vise YES.
    e. Vi har kun opplevd å få NO på Virtualization Enabled in Firmware. Dette må settes på i BIOS og det varierer fra maskin til maskin hvordan dette gjøres. Anbefaler å sjekke CPU eller security i BIOS på den enkelte maskinen.
3. Installere Hyper-V
    a. Åpne Turn windows features on/off
    b. Velg å installere Hyper-V
    c. Avslutt
4. Åpne Hyper-V Manager
    a. Trykk Windows-knapp og skriv inn *Hyper-V Manager*, trykk enter
    b. Velg Action-> Virtual Switch Manager
    c. Velg New Virtual Network Switch
    d. Klikk *External* og velg *Create Virtual Switch*
    e. Gi et navn til switchen og velge *External Network* og ditt nettverkskort, klikk OK
    f. Velg Action -> New -> Virtual Machine, klikk Next
    g. Gi den virtuelle maskinen et navn, klikk Next
    h. Angi hvor mye minne du ønsker å allokere (typisk 2 GB), klikk Next
    i. Velg switchen som ble opprettet i e), klikk Next
    j. Velg *Use an existing virtual hard disk*, klikk Browse og bla deg frem til mappen med imaget du har lastet ned og velg filen som ligger inne i mappen *Virtual Hard Disks*, klikk Next og Finish
    k. Høyreklikk på den virtuelle maskinen som er opprettet og velg Start
    l. Dobbeltklikk på den virtuelle maskinen
    m. Logg på med *Administrator* og passord *KjellTRing1*
    n. Verifisèr at du har internettilgang på den virtuelle maskinen
    o. Hent ut IP-adressen (f.eks ipconfig i cmd)
    p. Gå ut av den virtuelle maskinen og åpne en nettleser
    q. Octopus skal du nå finne på http://<ip-til-virtuell-maskin>:8081
    Brukernavn *admin* passord *KjellTRing1*
    r. Team City skal du nå finne på http://<ip-til-virtuell-maskin>:8080
    Brukernavn *admin* passord *KjellTRing1*
5. Hente kode
    a. I Explorer besøk \\<ip-til-virtuell-maskin>\git -> logg på med *administrator* og passord *KjellTRing1*
    b. Åpne Git-bash eller cmd
    c. Skriv inn git clone //<ip-til-virtuell-maskin>/git/WebApp1

Kom-i-gang (Uten Hyper-V image)

1. Installere Team City
    a. Installer DeploySoftware\TC TeamCity-8.0.4.exe
    b. Velg port 8080 (isteden for default 80)
    c. Velg at servicen skal kjøre som systembruker
    d. Bekreft ok hvis du får spørsmål om å endre firewall regler (Egen popup)
    e. Lag Administrator konto (admin/KjellTRing1)

2. Installere Octopus server
    a. Installer DeploySoftware\Octopus.1.6.3.1723.msi
    b. Hvis du får beskjed om at du må enable IIS-features, kjør kommandoen Octopus foreslår (i Administrasjons-kommandolinje), og så <check again> .Eksempel på kommando:



    c. Welcome to Octopus!
    d. Trykk <Next> til du kommer til Server
    e. Trykk <Install> - vent til servicen starter og status vises grønn (Running!)
    f. Trykk <Next> og skriv port 8081 *og* trykk <Create Site>, og Finish når den er startet.
    g. Åpne http://localhost:8081
    h. Lag Administrator konto (admin/KjellTRing1)
3. Hente kode
    a. Fork https://github.com/madsny/WebApp1/ og git clone lokalt

Oppgave: Bygg i Team City

I denne oppgaven skal vi sette opp et bygg i Team City som trigger på endringer som er gjort i git-repoet vi har klonet.

Oppsett av Bygg

1. Navigèr til Team City i browseren
2. Logg på med brukernavn og passord (admin/KjellTRing1)
3. Opprett et nytt prosjekt
4. Legg til *Build configuration* for det nye prosjektet
5. Konfigurer *VCS settings* til å gå mot Git repoet
6. Legg til et byggesteg som kan bygge prosjektet
    a. Hint: Add Build Step mot *.sln filen til løsningen
7. Verifisèr at prosjektet bygger ved å trykke *Run* på det opprettede prosjektet i Team City

Sette opp trigger for automatisk bygg ved endringer

1. Navigèr til Team City i browseren
2. Legg til en *Build Trigger* på prosjektet (hint: VCS trigger)
3. Commit en endring og se at bygget starter automatisk

Oppgave: Kjør tester

Det er mulig å kjøre tester som en del av byggestegene i Team City.

1. Legg til et nytt byggesteg som kjører testene i prosjektet
2. Sikre at bygget feiler dersom en av testene feiler (hint: Build Failure Conditions)

Oppgave: Lag NuGet pakke

For at man skal kunne rulle ut løsningen ved hjep av Octopus Deploy må man sørge for at løsningen man ønsker å deploye pakkes som en NuGet-pakke.

OctoPack (https://github.com/OctopusDeploy/OctoPack) er et sett med verktøy for å genrere NuGet pakke ihht kravene fra Octopus.

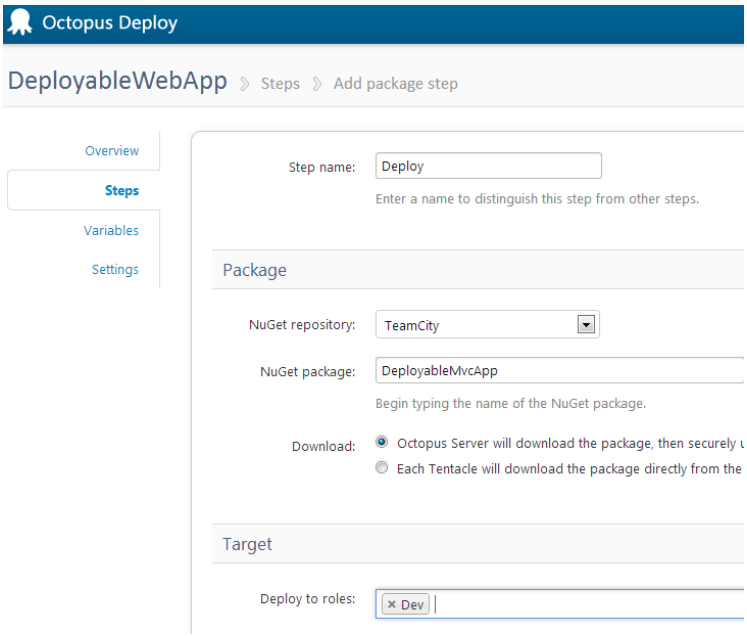
1. Endre byggekonfigurasjonen slik at Team City genrerere en slik pakke og legger den i C:\deploypackages på byggeserveren (hint: ta inn OctoPack i løsningen)
2. For at forskjellige bygg skal ha forskjellige versjonsnummer i Octopus, kan man bruke build.number i Team City (hint: General settings) til å endre på assemblyversjonen til det bygde prosjektet. OctoPack vil bruke assemblyversjonen som pakkeverson. (Hint: Assemblyinfo patcher)
3. Verifisèr at pakken blir generert og at neste pakke får et nytt versjonsnummer

Oppgave: Installere en Tentacle

IIS er installert på imaget og her er det satt opp en website med navn «MvcApplication» på port 17000.

For at Octopus skal kunne rulle ut løsningen til serveren trengs en såkalt Tentacle.

1. Bruk remote desktop inn på serveren
2. Gå til c:\downloads og dobbeltklikk på Octopus.Tentacle.1.6.3.1723.msi
3. Følg instruksjonene i veiviseren
4. Logg inn på Octopus serveren
5. Legg til et nytt miljø *Test*
6. Legg til Tentaclen som ble installert (hint: Add machine)
    a. Benytt Tentacle Admin Tool på serveren for utvekslingen av Thumbprint
7. Verifiser at Octopus får kontakt med Tentacle ved å kjøre en helsesjekk
8. I Octopus opprett et nytt prosjekt
9. Konfigurer et byggesteg i prosjektet som deployer pakken som blir bygget av Team City



Oppgave: Automatisk deploy

Vi ønsker at Octopus automatisk deployer ut ny versjon når det er sjekket inn endringer.
Benytt et av følgende to alternativ til å sette opp dette:

1. Benytte Octo.exe (https://github.com/OctopusDeploy/Octopus-Tools) og endre byggeoppsettet slik at Team City sier ifra til Octopus når bygget er ferdig og pakken er klar til å deployes. Octo.exe finnes også på imaget på c:\octopustools\octo.exe
2. Bruk Octopus-teamcity plugin (http://octopusdeploy.com/documentation/integration/teamcity). Installasjonsfilene ligger i c:\downloads på byggeserveren.

Oppgave: Promotering til Prod

Octopus har funksjonalitet for å gjennomføre transformasjoner (.NET Web config transformations) på alle *.config filer. Den vil automatisk kjøre transformasjoner som har samme navn som miljøet det deployes til.

Løsningen har en appsetting som heter «WebServiceUrl» som skal ha den samme verdien lokalt og i Test, men skal ha verdien http://prod.kuldataservice.no/api i Prod.

1. Lag en web.config transformasjon for å endre dette for Prod
   Syntax hint (http://msdn.microsoft.com/en-us/library/dd465326(v=vs.110).aspx)
   Nyttig utvidelse til Visual Studio ifm transformasjoner, SlowCheetah, finner dere her http://bit.ly/otg3b1
2. Opprett en ny site på serveren kallt *MvcApplicationProd* på port 18000 (husk app pool versjon til 4)
3. Opprett et nytt miljø i Octopus som benytter seg av den nye siten.
   Hint: Octopus har mange ferdigdefinerte variabler som blant annet kan instruere Octopus med navnet på siten i IIS (OctopusWebSiteName). For mer info om variabler se http://octopusdeploy.com/documentation/features/variables.
4. Forsøk å trigge en ny utrulling til Dev enten ved å gjøre endring i koden, eller ved Run i Team City
5. Når utrulling til Test er komplett, bruk *Promote to…* funksjonaliteten i Octopus for å prøve å rulle ut til Prod.
6. Verifiser at utrullingen har fungert og besøk Prod på http://<ip-til-virtuell-maskin>:18000
7. Url'en http://prod.kuldataservice.no/api bør vises på siden.