



Projectwerk - Verslag



Administration

Jorn Jambers, Wout Decré & Jan Collijs
3 TI
juni 2009

Katholieke Hogeschool Leuven
Departement Gezondheidszorg en Technologie
Herestraat 49 - 3000 LEUVEN
Tel. +32 16 37 52 00
Fax +32 16 37 52 99
gt@khleuven.be

INHOUDSOPGAVE

Inhoudsopgave	1
Inleiding	3
Omschrijving	3
Puppet	4
Monitoring	4
Scripting	5
Python	5
Kenmerken Python	5
Databank	6
MySQL	6
Kenmerken MySQL	6
Kickstart	8
Puppet	9
Configuratie (Manifestes)	9
Master	9
Nodes	10
Nagios	11
Inleiding	11
Printscreen	12
Scripts	13

mysql-to-nagios.py	13
mysql-to-puppet.py	13
puppetman.py	14
Slot	15
Bronnen	16

Inleiding

In het kader van het projectwerk aan de Katholieke Hogeschool Leuven in de richting Toegepaste Informatica, werkten wij aan een systeem dat met behulp van het Open Source project Puppet de automatisatie van een serverpark op zich neemt. Dankzij het door ons uitgewerkte systeem wordt het mogelijk om, met behulp van verschillende scripts en gebruikmakende van Kickstart, een server, met een minimale inspanning en binnen een zo beperkt mogelijke tijdsperiode, binnen de opstelling te krijgen.

In dit document vindt u een algemeen beeld terug van de door ons gebruikte technieken, evenals een antwoord op de vraag waarom wij voor deze technieken gekozen hebben.

We wijzen u er ook graag op dat er een tweede document bestaat waarin de verschillende stappen van de opstelling tot in detail beschreven worden en waardoor u in de mogelijkheid zou moeten verkeren om op zelfstandige basis snel een testopstelling op te zetten.

Wij wensen u veel plezier met het doornemen van dit document en natuurlijk ook met Puppet.

Omschrijving

Het onderhoud van een serverpark brengt veel werk met zich mee. Een systeembeheerder dient evenwel tijdens dit onderhoud een aantal taken te herhalen. Hoe vaak deze taken herhaald dienen te worden is natuurlijk afhankelijk van de grootte en het doel van het netwerk.

Wij hebben getracht om deze taken van een systeembeheerder te minimaliseren door in de verschillende stappen van installatie, configuratie, en onderhoud zoveel

mogelijk automatisatie aan te brengen. Op die manier komt er meer tijd vrij voor de systeembeheerder die hij of zij dan aan nuttigere zaken kan besteden, in plaats van altijd dezelfde handelingen te moeten herhalen.

Om dit verhaal tot een goed einde te brengen, hebben wij ervoor gekozen om gebruik te maken van Puppet. Samen met verschillende scripts, een databank en Kickstart zou het mogelijk moeten zijn om een server in minder dan 30 minuten operationeel te krijgen binnen een netwerk.

Ready, steady, GO!

Puppet

Puppet is een Open Source pakket dat gratis aangeboden wordt door Reductive Labs. Het is gebaseerd op Ruby en het heeft tot doel om administratieve taken, die via het gekende client-server model gedistribueerd worden, op een centrale manier af te handelen. Puppet kan overweg met alle UNIX-based systems, wat voor zeer veel serveromgevingen een groot pluspunt is.

Binnen onze opstelling maken wij gebruik van dit Open Source pakket om de configuraties van de verschillende servers op een geautomatiseerde manier te verspreiden over het netwerk.

Monitoring

Een monitorsysteem wordt binnen een bedrijf gebruikt om het opvolgen van de status van verschillende systemen te vergemakkelijken. Door hiervan gebruik te maken kan er immers sneller ingespeeld worden op eventuele problemen.

Het geeft in de eerste plaats een overzicht van de status van de volledige infrastructuur weer en dit met het oog op het vlotter laten verlopen van het onderhoud ervan. Er zijn zelfs enkele monitorsystemen in staat om bepaalde criteria te controleren en indien nodig (bijvoorbeeld bij het overschrijden van dergelijke criteria) zelf in te grijpen om te voorkomen dat er problemen op het systeem ontstaan.

Binnen dit projectwerk maken wij gebruik van het Open Source project Nagios, waar we verder in dit werk dieper op zullen ingaan .

Scripting

Door gebruik te maken van scripts kunnen we bepaalde handelingen, met behulp van kleine programma's, sneller en gebruiksvriendelijker laten uitvoeren. Enkele van deze scripts worden ook automatisch uitgevoerd, wat maakt dat er op die manier weer enkele taken van de systeembeheerder overgenomen worden.

Op een volgens ons gebruiksvriendelijke manier scripten we erop los met behulp van Python, een taal die ondersteund wordt door de meeste UNIX-omgevingen. Daardoor zal de samenwerking met Puppet zo optimaal mogelijk kunnen gebeuren.

Python

Python is een scripttaal die ontwikkeld werd begin jaren 90. Deze taal is een high-level, 'gestructureerde', Open Source programmeertaal die voor een groot aantal verschillende programmeertaken kan worden ingezet. Zo kan ze gebruikt worden voor zowel relatief eenvoudige scripts, als voor meer complexe applicaties.

Kenmerken python

Python is een 'geïnterpreteerde' programmeertaal die automatisch in bytecode wordt gecompileerd op het moment van uitvoeren. Deze bytecode wordt eveneens automatisch op disk opgeslagen zodat compilatie niet opnieuw hoeft plaats te vinden, tenzij de bron opnieuw aangepast of veranderd wordt. Het is ook een zogenaamde 'dynamically typed language' met object georiënteerde features en constructs, alhoewel je deze niet hoeft te gebruiken.

Een van de meeste unieke en in het oog springende aspecten van Python is dat whitespace een wezenlijk onderdeel uitmaakt van de syntax. In plaats van 'block delimiters' (haakjes over het algemeen) wordt indentatie gebruikt om aan te geven waar de blokken code beginnen en eindigen.

Databank

Om de gegevens van de verschillende servers (nodes) binnen de opstelling te bewaren hebben wij ervoor geopteerd om gebruik te maken van een MySQL-databank. Op die manier gebruiken we voor alle configuraties, zowel voor het Puppet pakket als voor de Nagios monitoring tool, dezelfde gegevens.

Het gebruik maken van een databank heeft als voordeel dat, als er gegevens gewijzigd worden, alle gebruikte systemen toch nog beschikken over de laatste versie ervan. Met behulp van de scripts geschreven in Python, is het ook mogelijk om deze gegevens op een gebruiksvriendelijke manier te benaderen, aan te passen, te creëren, en te verwijderen uit deze databank.

De databank zou eventueel ook vervangen kunnen worden door LDAP. Aangezien we reeds de handen vol hadden met de implementatie van de huidige opstelling en er daarnaast ook beperkingen kunnen optreden wanneer er gebruik gemaakt wordt van de standaardconfiguratie van LDAP, hebben we er evenwel voor geopteerd om deze, toch wel complexe omgeving links te laten liggen.

MySQL

MySQL is een Open Source relationeel databasemanagementsysteem (RDBMS) dat gebruik maakt van SQL (Structured Query Language). MySQL werd vroeger vooral gebruikt voor toepassingen zoals fora en gastenboeken, meestal in combinatie met PHP.

Tegenwoordig is het de basis van een breed scala aan internettoepassingen, maar ook van stand-alone software.

Kenmerken MySQL

MySQL bestaat uit een verzameling clientprogramma's, zoals mysql en mysqldump waarmee automatisch of interactief met de server gecommuniceerd kan worden.

Het is een populair databasemanagementsysteem dat voor het gestructureerd opslaan van gegevens gebruikt wordt. Dit systeem wordt voor vele toepassingen gebruikt. Een voorbeeld van websites die gebruik maken van MySQL zijn de

sites van Wikipedia, de vrije encyclopedie. Een bekende MySQL-frontend is dan weer phpMyAdmin. Dit is een webgebaseerd data administratie programma geschreven in PHP.

Kickstart

Het verhaal van automatisatie en tijdswinst klinkt zeer mooi, maar wanneer de systeembeheerder bij de installatie van het besturingssysteem nog steeds de verschillende stappen van de installatieprocedure dient te doorlopen hebben we er niet veel aan.

Om hier een oplossing voor te bieden, kozen we ervoor om gebruik te maken van Kickstart. Kickstart biedt je de mogelijkheid om de antwoorden op de verschillende vragen van het installatieproces in een bestand op te slaan. Bij de eigenlijke opstart zal het installatieproces de antwoorden uit dit bestand gebruiken om het systeem naar jouw wensen in te stellen. Op die manier komt er opnieuw tijd vrij die een systeembeheerder op andere manier kan invullen.

We zijn er in geslaagd om de installatie van het Puppet packet volledig in deze Kickstart-bestanden mee te geven. Op die manier wordt het mogelijk om met behulp van een cd-rom zowel de Puppet master als de Puppet slave te installeren.

Zo dient de systeembeheerder achteraf nog enkel met behulp van het beheerscript (op de Puppet master) aan te geven welke node (slave) welke configuratie dient te krijgen, bv. Webserver, waarna de nodige stappen door Puppet uitgevoerd worden zonder manuele tussenkomst om van de node in kwestie een operationele webserver te maken.

Puppet

Configuratie (Manifests)

De Puppet-configuratie bestaat uit verschillende manifests-bestanden. Van hieruit kan je nodes (Puppet clients) toevoegen en ze een bepaalde klasse (vb. webserver) toekennen.

Een klasse is een verzameling van bestanden, mappen, services, gebruikers, groepen,... Zo bestaat de klasse webserver uit de service apache2, verschillende Apache configuraties, de "www" gebruiker en groep, ...

Je kan al je service-configuratiebestanden dus bijhouden en beheren op je Puppet master en ze van hieruit doorsturen naar je nodes. Deze bestanden kunnen gewoon plain tekstbestanden zijn of Puppet templates die vanuit de manifests ingevuld worden (vb. de Virtual Hosts van de Apache configuratie).

Binnen onze opstelling werden er manifests geschreven eigen aan het project om op een zo eenvoudig mogelijke manier de gebruiker toe te laten een webserver op te stellen zonder zich zorgen te hoeven maken over deze manifests.

Master

De Puppet master houdt de verschillende manifests bij. Er draait een daemon waarop alle Puppet slaves op regelmatige tijdstippen connectie maken. Deze downloaden dan hun configuratie en vergelijken deze met de huidige eigen configuratie. Indien er verschillen zijn zullen de nieuwe aanpassingen uitgevoerd worden door de Puppet slave.

Nodes

Onder node verstaan we in dit project een server of werkstation die als Puppet slave geconfigureerd werd en de instructies van de Puppet master volgt.

Nagios

Nagios werd gekozen omwille van zijn configuratiewijze. Met behulp van scripts zijn we erin geslaagd om de Nagios configuratiebestanden automatisch te genereren waardoor deze gebruik maken van dezelfde gegevens als deze waarvan Puppet gebruik maakt.

Zo ontstond de mogelijkheid om met een simpel commando aan te geven of een server al dan niet een Nagios monitoring server wordt die quasi onmiddellijk beschikt over de juiste configuratie zonder manuele tussenkomst.

Inleiding

Nagios is een populair Open Source computersysteem en netwerksurveillance applicatie. Het houdt servers en services in de gaten die je specificeert en het stuurt berichten of mails wanneer er dingen stuk gaan en wanneer de services terug hersteld zijn.

Nagios ontstond onder de naam NetSaint, dat geschreven is en onderhouden wordt door Ethan Galstad samen met een groep ontwikkelaars. Deze houden Nagios actief en voorzien plugins van updates.

Printscreen



Scripts

mysql-to-nagios.py

Zoals hierboven reeds aangehaald werd, wordt de configuratie van Nagios op een automatische wijze aangemaakt. Met behulp van het mysql-to-nagios script worden de nodige gegevens uit de centrale databank gehaald, waarna ze in de juiste vorm gegoten worden zodanig dat Nagios ze kan lezen en gebruiken.

Dit gebeurt (zonder tussenkomst van de systeembeheerder) op automatische wijze of nadat er gegevens gewijzigd werden met behulp van het puppetman script of op regelmatige tijdstippen met behulp van een cron job gekend binnen de UNIX-omgeving.

Deze aangemaakte bestanden kunnen dan door Puppet worden verzonden naar de juiste node, met name diegene die als Nagios monitoring server aangeduid werd.

mysql-to-puppet.py

Dit script zal voor iedere node uit de centrale databank de benodigde manifests samenstellen. Deze kunnen daarna, met behulp van het puppet-gebeuren, doorgegeven naar de specifieke node.

Wederom gebeurt dit op een automatische manier of nadat er gegevens werden gewijzigd met behulp van het puppetman script of op regelmatige tijdstippen met behulp van een cron job gekend binnen de UNIX-omgeving.

puppetman.py

Met behulp van het puppetman script is het mogelijk voor de gebruiker om op een zeer gebruiksvriendelijke manier de zonet geïnstalleerde nodes te configureren met enkele simpele commando's.

De aanpassingen die gemaakt worden met behulp van dit script worden opgeslagen in de centrale databank, waarna de voorgaande scripts hun taken uitvoeren en waardoor het systeem zich aanpast aan de gewijzigde situatie.

Kortom, het is de brandstof nodig voor de motor (databank) van de hele opstelling.

Slot

De opbouw van dit project heeft zeer veel bloed, zweet, en tranen van ons allen gevegd. Maar nu het allemaal in een duidelijk geheel naar voren gebracht kan worden, heeft er zich toch een gevoel van trots zich meester gemaakt.

Het doorgronden van Puppet, het verstaan van Nagios, het gebruik van toch wel enkele complexe zaken binnen Python en het schrijven van alle documentatie, zowel op de Wiki als met behulp van Latex, De meeste zaken waren zo goed als ongekend voor ons en eerlijkheidshalve kunnen we niet anders dan toegeven dat we nog steeds niet alles tot in detail onder de knie hebben. Al staan we er toch van versteld in welke korte termijn we toch wel een zeer gedegen kennis van deze verschillende aspecten opgebouwd hebben.

Ook de samenwerking tussen de verschillende teamleden verliep meer dan vlot, al heeft dit waarschijnlijk wel te maken met het feit dat we elkaar wel al langer kenden en wisten wat we aan elkaar gingen hebben.

Het was een vermoeiende en toch wel op sommige punten stresserende periode, maar de voldoening is enorm.

” Master of Puppets I’m pulling your strings
Twisting your mind, smashing your dreams
Blinded by me, you can’t see a thing
Just call my name, ’cause I’ll hear you scream
Master
Master”¹

¹Master of Puppets by Metallica

Bronnen

- Pulling Strings with Puppet door James Turnbull
- Puppet website
- Wikipedia - MySQL
- Wikipedia - Python
- Wikipedia - Kickstart
- Nagios