



HoGent

Faculteit Bedrijf en Organisatie

Security en Managebility van Docker containers

Tomas Vercautter

Scriptie voorgedragen tot het bekomen van de graad van
Bachelor in de toegepaste informatica

Promotor:
Bert Van Vreckem
Co-promotor:
Bert Van Vreckem

Instelling: —

Academiejaar: 2015-2016

Tweede examenperiode

Faculteit Bedrijf en Organisatie

Security en Managebility van Docker containers

Tomas Vercautter

Scriptie voorgedragen tot het bekomen van de graad van
Bachelor in de toegepaste informatica

Promotor:
Bert Van Vreckem
Co-promotor:
Bert Van Vreckem

Instelling: —

Academiejaar: 2015-2016

Tweede examenperiode

Samenvatting

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Voorwoord

Inhoudsopgave

1	Inleiding	4
1.1	Probleemstelling en Onderzoeksvragen	4
2	Methodologie	5
I	Corpus	6
3	Security	7
3.1	Host Configuration	7
3.1.1	Up to date houden van het systeem	8
3.1.2	Configureren van het systeem	8
3.1.3	Auditen van het systeem	9
4	Manageability	10
5	Logging	11
6	Conclusie	12

Hoofdstuk 1

Inleiding

Docker containers is een manier van virtualiseren van services. Hierbij worden services in een

In deze bachelorproef zal ik onderzoek doen naar de security en managebility van Docker containers en deze vergelijken met de security en managebility van het draaien van de services reeks op de virtuele machine.

Deze bachelorproef is onderverdeeld in drie grote delen. In de eerste plaats bespreek ik de security zelf. In een tweede deel zal ik het hebben over de managebility van Docker containers. En als laatste onderdeel bespreek ik logging bij docker containers.

1.1 Probleemstelling en Onderzoeksvragen

Hoofdstuk 2

Methodologie

Deel I

Corpus

Hoofdstuk 3

Security

Op vlak van security ben ik gestart van CIS Docker 1.6 Benchmark geschreven door Pravin Goyal (2015). Hierin is in samenwerking met de "Center for Internet Security-onderzoek gedaan naar de beveiliging van docker containers. Dit onderzoek is onderverdeeld in zes grote onderdelen namelijk Host configuration, Docker daemon configuration, docker daemon configuration files, Container images and build file, Container runtime en docker security operations. Hierbij wordt er gekeken welke van deze elementen standaard in docker worden uitgevoerd. In dit hoofdstuk zal ik dezelfde zes onderdelen behandelen en vergelijken. Hierbij zal ik kijken hoe we deze beveiliging aanpakken wanneer we onze services in een Docker container draaien tegenover wanneer we deze native in onze virtuele machine draaien.

Hierbij heb ik ook telkens deze methodes voor Docker containers uitgetest om te kijken hoe makkelijk deze kunnen geïmplementeerd worden in een nieuwe omgeving of bestaande omgeving. Een grote hulp hiervoor was de github repository "docker-bench-security" van docker een grote hulp. Dit is een script die alle elementen die worden aangehaald in CIS Docker 1.6 Benchmark automatisch getest en feedback over gegeven.

3.1 Host Configuration

Een van de belangrijkste onderdelen in het beveiligen van de Docker stack is het beveiligen van de host machine waarop de docker containers zich zullen bevinden. Dit heeft drie onderdelen namelijk het up to date houden van het systeem, het juist

configureren van het systeem, en het auditten van het systeem.

3.1.1 Up to date houden van het systeem

Om de veiligheid van een systeem te kunnen garanderen moet er ten allen tijde gezorgd worden dat het systeem up to date blijft. Dit zorgt ervoor dat security flaws die gekend en gepatcht zijn ook op het systeem beveiligd zijn. Doordat we Docker gebruiken komt hier nog een extra laag bij. Bij het Updaten van een machine waarop de services rechtstreeks draaien moet enkel het host systeem, de kernel van het host systeem en de services up to date blijven. Doordat Docker een kernel deelt met het hostsysteem wordt het up to date houden van de kernel belangrijker. Naast het onderhouden van de voorvernoemde elementen moet er nu, doordat we Docker gebruiken, nog met twee extra dingen rekening gehouden worden. Enerzijds moet Docker zelf nu ook upgedate worden, maar een onderdeel die soms vergeten wordt is dat elke container nu ook zijn eigen besturingssysteem gebruikt. Hierover wordt er verder bij het beveiligen van Container images meer uitleg over gegeven.

3.1.2 Configureren van het systeem

Bij het configureren van het hostsysteem wordt net zoals bij het up to date houden van het systeem dezelfde stappen van een native applicatie op een hostsysteem configureren gevolgd. Maar door de extra laag komen daar nog extra stappen bij. Bij een systeem waar Docker op draait wordt er aangeraden om een aparte partitie te creëren voor de containers. Dit is een simpele operatie waardoor alle docker gerelateerde files zich niet meer tussen de files van het hostsysteem bevinden.

Daarnaast moet ook onder controle gehouden worden wie toegang krijgt tot de docker daemon. De Docker daemon heeft 'root' access, gebruikers die worden toegevoegd aan de 'docker' usergroup en geen 'root' privileges hebben kunnen hierdoor 'root' access verkrijgen tot het hostsysteem. Men directories sharen tussen het host systeem en een guest container. Doordat containers standaard altijd runnen als 'root' heeft de container als de '/' directory gemount is op deze container ongelimiteerde toegang tot het volledige host systeem. Hierdoor kunnen zonder restricties aanpassingen aan het hostsysteem gedaan worden. Dit betekent dat een gebruiker die toegang heeft tot de docker daemon verhoogde rechten kan verkrijgen door gewoon een container op te starten.

Zoals bij een virtuele machine waarop de services rechtstreeks wordt ook bij een

systeem waar docker containers op draaien aangeraden overbodige services uit te schakelen of te verwijderen. Moest een gebruiker door een van deze services toegang krijgen tot het hostsysteem. Zou hij makkelijker 'root' rechten kunnen vergrijpen volgens de manier beschreven in de vorige paragraaf.

3.1.3 Auditen van het systeem

Hoofdstuk 4

Manageability

Hoofdstuk 5

Logging

Logging is een zeer belangrijk onderdeel van docker containers. Omdat we onze docker containers veel meer gaan isoleren en onze containers veel vluchtiger zijn dan gewone services op een virtuele host wordt loggen van

Hoofdstuk 6

Conclusie

Bibliografie

- Anderson, C. (2015). Docker [Software engineering]. *IEEE Software*, 32(3):102–c3.
- Brockmeier, J. (2014). Is It Safe? A Look at Docker and Security from LinuxCon — Project Atomic.
- Docker (2015). Introduction to Container Security. Technical report.
- Docker (2016). Docker security.
- Ewindisch (2014). On the Security of Containers.
- Jessie Frazelle (2016). Docker Engine 1.10 Security Improvements.
- Lenny Zeltser (2015). Security Risks and Benefits of Docker Application Containers.
- Mónica, D. (2015). Understanding Docker Security and Best Practices.
- Mouat, A. (2015). *Docker Security*.
- Petazzoni, J. (2013). Containers & Docker: How Secure Are They?
- Pravin Goyal (2015). CIS Docker 1.6 Benchmark v1.0.0. Technical report.
- Toni de la Fuente (2015). Docker Security Tools: Audit and Vulnerability Assessment.
- Voelker, A. (2015). Why I don't use Docker much anymore - Abe Voelker.

Lijst van figuren

Lijst van tabellen