



Ymyr Osman

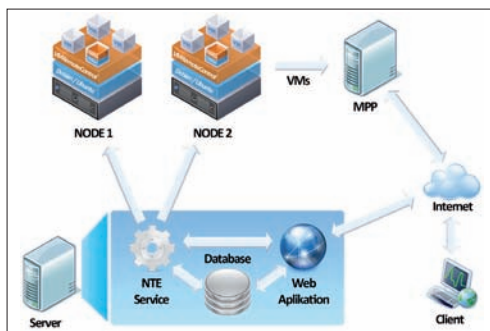


Oliver Zürcher

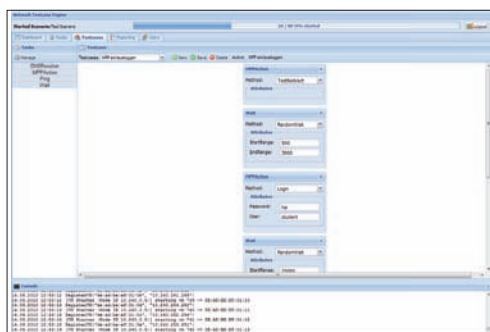
Diplomanden	Ymyr Osman, Oliver Zürcher
Examinator	Prof. Beat Stettler
Experte	Markus Vöglin, Clounet AG, Arlesheim BL
Themengebiet	Internet-Technologien und -Anwendungen

Network Testcase Engine

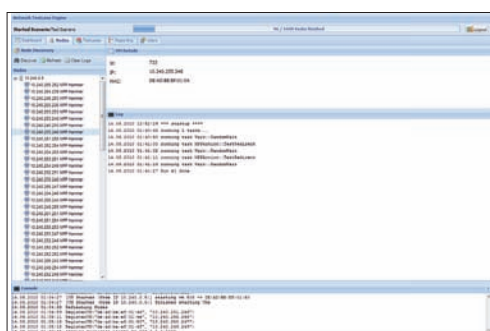
16



Ausgangslage: Das INS hat 2003 für den Flughafen Zürich die Internet-Access-Plattform MPP entwickelt. Seither hat sich der Kundenstamm auf Hochschulen, Grossfirmen und Serviceprovider vergrössert. Die Anzahl Nutzer dieser Plattform ist in den letzten Jahren regelrecht explodiert, was zu einem starken Anstieg der Last auf den MPP-Plattformen der Hotspots geführt hat. Ein detailliertes Testen des entwickelten Programmcodes ist daher sehr wichtig. Die Anzahl Clients für ein der Realität entsprechendes Szenario ist aber bereits zu gross, um das Testen von Hand durchführen zu können. In der Semesterarbeit «Captive Portal Load Generator» wurde ein Konzept zur Simulation von Tausenden von virtuellen Clients sowie ein erster lauffähiger Prototyp entwickelt. Dieser soll nun erweitert werden.



Vorgehen/Technologien: In der Vorarbeit erwies sich die Kernel-Based Virtual Machine (KVM) als optimale Lösung für die Virtualisation vieler Clients. Deshalb kommt auch bei der Network Testcase Engine KVM zum Einsatz. In einer ersten Analyse wurden die verschiedenen möglichen Tasks definiert und die Architektur festgelegt. Als Webserver kommt ein Windows-Server 2008 R2 zum Einsatz. Für die Persistierung wird ein Microsoft SQL 2008 Server eingesetzt. Als OR-Mapper wird das neue Entity Framework 4 CTP3 verwendet. Die Webseite wird in ASP.NET programmiert, wobei der grösste Teil des Codes in Javascript geschrieben wird. Zum Einsatz kommt hier das Javascript-Framework ExtJS. Auf dem Linux-Rechner kommt .NET 3.5 unter der Mono Runtime zum Einsatz. Für die Interprozesskommunikation (IPC) wird auf das .NET Remoting Framework zugegriffen, welches seit .NET 2.0 Bestandteil des Frameworks ist und somit auch unter Mono funktioniert.



Ergebnis: Die Network Testcase Engine wird über ein zentrales Web Frontend gesteuert. Testcases können aus einzelnen Tasks per Drag&Drop zusammengestellt und anschliessend in Szenarien eingebettet werden. Beim Start eines Szenarios werden automatisch genügend VMs auf die KVM-Hosts verteilt und aufgestartet. Nach dem Startvorgang können die VMs im Webinterface angewiesen werden, ihre Testcases auszuführen. Dazu wird ein Multicast-Paket verschickt, um allen VMs das Signal möglichst gleichzeitig zu geben. Während ein Szenario läuft, können die Logs der virtuellen Maschinen überprüft werden. Im Tab Reports werden die Ergebnisse von Szenariodurchläufen aufgelistet.