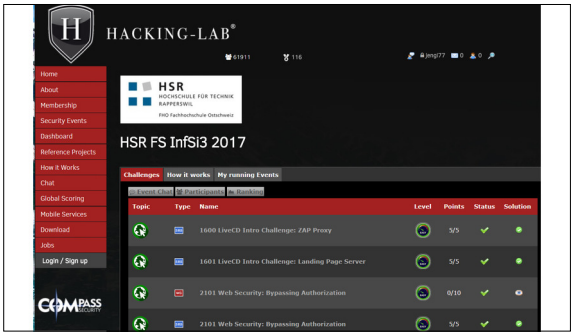
Yanick Gubler

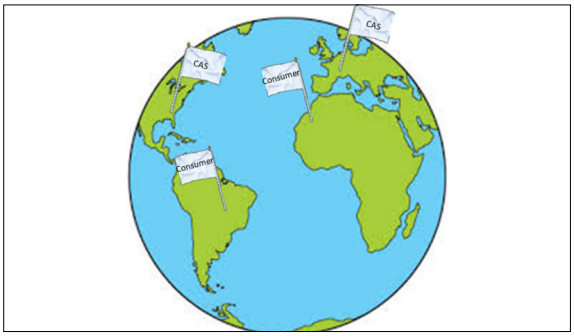
Janick Engeler

Studenten	Yanick Gubler, Janick Engeler
Examinator	Ivan Bütler
Themengebiet	Sicherheit
Projektpartner	Security Competence GmbH, Jona, St. Gallen

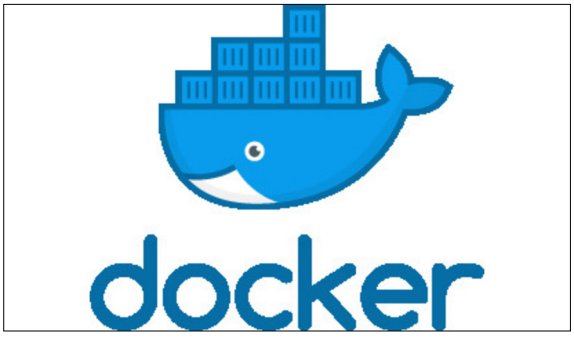
Hacking-Lab 2.0



Hacking-Lab Now



Hacking-Lab 2.0



Docker Symbol

Ausgangslage: Die Security Competence GmbH möchte ein Konzept für ein neues Hacking-Lab 2.0 ausarbeiten und dazu einen ersten Prototyp erstellen. Das jetzige Hacking-Lab dient dazu Hacking Aufgaben zu lösen und wird vor allem in der HSR zum praktischen Lernen der Informationssicherheit und bei Compass Security für spezielle Events verwendet. Probleme des jetzigen Systems sind unter anderem, dass das bestehende User Interface weder zeitgemäss noch mobilefähig ist, es nur einsprachig betrieben werden kann, es nicht einfach erweiterbar ist und so keine Skalierbarkeit gewährleistet ist. Darüber hinaus generiert es für die Betreiber einen grossen Mehraufwand, weil erstens laufend neue Challenges erfasst und bearbeitet werden müssen und zweitens die Lösungen der durchgeführten Übungen mehrheitlich von Hand korrigiert werden. Das jetzige System basiert auf einer sogenannten Live-CD (beinhaltet Hackingumgebung und /-tools), was einen unnötigen Aufwand für den Endanwender bedeutet.

Ziel der Arbeit: Das bestehende System soll nach und nach durch ein neues ersetzt werden. Für das neue System wird der Fokus auf eine weltweite Nutzung gelegt. Dies führt zu einer neuen Problematik, da z.B. Länder wie China ihren verschlüsselten Internetverkehr drosseln und es somit schwierig wird aus diesem Land auf Server anderer Länder zuzugreifen. Dies würde zu Performance Engpässen führen und so eine sinnvolle Nutzung verhindern. Deshalb wird eine hohe Skalierbarkeit angestrebt, damit mehrere Instanzen des Hacking-Lab 2.0 parallel betrieben werden können und der Verkehr über die Landesgrenzen minimal gehalten werden kann. Eine weltweite Nutzung bedeutet auch, dass das System mehrsprachig betrieben werden können soll. Ein entscheidender Punkt ist der Mehraufwand. Diesem Problem soll dadurch begegnet werden, dass eine Community aufgebaut wird, welche selbstständig neue Challenges und Musterlösungen erfasst. Dadurch können auch die Übersetzungen direkt von einem Muttersprachler erstellt werden. Damit die Qualität der angebotenen Inhalte nicht unter dem Community-Gedanken leidet, soll ein Review-System implementiert werden, wodurch neue oder veränderte Challenges durch einen Nutzer mit entsprechenden Rechten kontrolliert und freigegeben werden müssen, bevor sie zugänglich sind. Damit die Live-CD des bestehenden Hacking-Labs in der Version 2.0 nicht mehr benötigt wird, sollen die benötigten Ressourcen (Dockercontainer, VM's u.ä.) direkt einer Challenge zugewiesen werden können.

Ergebnis: Das Ergebnis dieser SA soll ein intelligentes Konzept sein, das alle zuvor genannten Probleme sinnvoll abdeckt bzw. verhindert. Deshalb wurde die erste Hälfte der Zeit dafür verwendet, dieses Konzept im Gespräch mit dem Betreuer zu erarbeiten. Nach der konzeptuellen Ausarbeitung wurde ein erster Prototyp des Challenge Authoring System (CAS) implementiert, der für die anderen verwendeten Systeme ein REST-API anbietet, um dort auf einfache Weise die Daten beziehen zu können. Das CAS besteht aus drei Komponenten.

- CAS-Client welcher das GUI für die Challenge Erstellung und Übersetzung liefert
- CAS-Server welcher die Daten verwaltet und das API anbietet
- CAS-mysql welches die Datenbank hält

Alle Subsysteme werden in einem eigenen Dockercontainer implementiert. So kann schnell und einfach ein neues CAS hochgefahren werden, indem man lediglich das GIT-Repo klont und die enthaltenen Skripts ausführt.