Technische Universität Dresden

Fakultät Informatik

Professur für Datenschutz und Datensicherheit

Projekt HoneySens

6. Zwischenbericht

Forschungsprojekt zur Entwicklung eines Sensornetzwerks, um Angriffe auf die IT-Infrastruktur des Freistaates zu erkennen

Stand: 14.09.2016

Autor: Dipl.-Inf. Pascal Brückner
Betreuung: Dr.-Ing. Stefan Köpsell

1 Einleitung

Der Beauftragte für Informationssicherheit des Landes Sachsen prüft zur Verbesserung der IT-Sicherheit des Sächsischen Verwaltungsnetzes (SVN) die Grundlagen zum Aufbau eines Sensornetzwerkes, mit dem verdächtige Zugriffe auf Netzwerkdienste und -geräte erkannt und zuständige Stellen zeitnah informiert werden können. Das geplante System soll sich insbesondere durch eine leichte Bedienbarkeit und gute Skalierbarkeit auszeichnen. Hauptaugenmerk liegt dabei auf den Funktionalitäten, die die Wartung und Verwaltung der Sensorinfrastruktur betreffen. Weitere Faktoren, darunter die möglichst einfache und transparente Integration der zu entwerfenden Architektur in das bestehende Netzwerk, sowie ein autonomer, von anderen Netzwerkkomponenten und -diensten unabhängiger Betrieb sind ebenfalls zu berücksichtigen. Ein weiterer wichtiger Bestandteil des Forschungsprojektes ist zudem die Koordinierung der vielfältigen Anforderungen der am Projekt interessierten und teilnehmenden Ressorts der Landesverwaltung.

In der vorhergehenden Projektphase hat die Web-Oberfläche starke optische und auch funktionale Veränderungen erfahren, um die Benutzbarkeit des Systems für Administratoren zu vereinfachen. Aufgrund der Vielfalt der Neuerungen war der Versionssprung auf 0.2.0 für das diese Änderungen enthaltende Release vorgesehen. Inhalt des aktuellen Projektabschnittes war nun die Finalisierung dieser neuen Version, was in umfassenden Test- und Bugfixing-Sessions mündete. Neben der Entfernung etlicher ungenutzter Altlasten aus dem Code wurde zudem auch die SSH-Honeypot-Unterstützung für die BeagleBone-Plattform reaktiviert und die Web-Oberfläche zur Sensorkonfiguration optisch und inhaltlich überarbeitet und an den aktuellen Projektstand angepasst. Aus den Labortests haben sich zudem noch eine Reihe weiterer notwendiger Verbesserungen für einen reibungslosen Deployment-Prozess ergeben. Somit war sichergestellt, dass auch die bisher bestehenden Test-Installationen problemlos auf die neue Sensor-Firmware und die neue Server-Revision aktualisiert werden können.

Das tatsächliche Update der bestehenden Installationen und die Ausweitung der Testphase an der TU Dresden war ebenfalls Bestandteil der Projektphase und verlief in allen Fällen problemlos.

Die im ersten Zwischenbericht beschriebenen Anforderungen behalten ihre Gültigkeit und gelten weiterhin als Richtlinie für zukünftige Entwicklungen. Sie werden an dieser Stelle nicht erneut ausgeführt.

2 Sachstand

Dieser Abschnitt gibt einen Überblick über die vom 19. Juli 2016 bis zum 18. September 2016 verzeichneten Fortschritte am HoneySens-Projekt. Teilbereiche, die in den Anforderungen des ersten Projektberichtes genannt, aber hier nicht weiter beschrieben werden, sind Gegenstand der zukünftigen Weiterentwicklung.

Reaktivierung des SSH-Honeypots Kryptographische Verbesserungen am SSH-Protokoll in aktuellen Versionen der verbreiteten Software *OpenSSH* hatten zur Folge, dass auch der als Teil der HoneySens-Firmware ausgelieferte Medium-Interaction-SSH-Honeypot *kippo* einer Aktualisierung bedurfte. Andernfalls wäre nur noch eine erfolgreiche Verbindung von veralteten SSH-Clients seitens des Honeypots akzeptiert worden. Aktuelle Varianten der Software erzeugten aufgrund der genannten stärkeren Kryptographie allerdings bereits beim Verbindungsaufbau immensen Rechenaufwand und führten schließlich dazu, dass der Kontakt zum kippo-Dienst auf den von uns genutzten BeagleBone-Geräten einem DoS-Angriff gleichkam. Dies war Anlass dafür, diesen Honeypot-Dienst zunächst für den Probebetrieb zu deaktivieren und nach einer Ausweichlösung zu suchen.

Inzwischen ist ein aktiv weiterentwickelter Fork des kippo-Projektes mit dem Namen *cowrie* entstanden, in dessen Bugtracker das genannte Problem ebenfalls diskutiert wurde. Die Entwickler hatten bereits Optimierungen vorgenommen, um den Verbindungsaufbau auf leistungsschwachen Geräten zu beschleunigen. Unsere eigenen Tests zeigten jedoch, dass noch immer eine Wartezeit von ca. 30 Sekunden bei jedem Verbindungsaufbau zum SSH-Honeypot auf den BeagleBone-Geräten entstand. Als Lösung des Problems erwieß sich schließlich die Installation der *GMP*-Bibliothek und ihres entsprechenden Python-Pendants *gmpy2*, die eine Vielzahl schneller mathematischer Operationen bereitstellt. Nach dieser Änderung war der Kontakt zum und die Interaktion mit dem SSH-Dienst wieder wie gewohnt möglich. Kleinere Anpassungen des Logging-Moduls für *cowrie* stellten die Kompatibilität der OpenSource-Lösung mit dem HoneySens-Server her, weswegen der Dienst nun wieder in allen Testnetzen aktiviert wurde.

Überarbeitete Sensorkonfiguration Die Masken im Webfrontend zur Konfiguration der Sensoren und ihrer Dienste wurden während dieser Projektphase nach dem Vorbild aller anderen Masken modularisiert und dem graphischen Stil der restlichen Anwendung angepasst 1. Die zur Verfügung stehenden Optionen werden dem Anwender nun platzsparender und damit auch übersichtlicher präsentiert. Weiterhin sind die jeweils zur Verfügung stehenden Dienste je nach Status (aktiviert oder deaktiviert) farblich eindeutig hervorgehoben und mit einem kurzen Beschreibungstext zum Zweck der entsprechenden Komponente versehen. Identische Veränderungen erfuhr ebenfalls der Dialog zum Hinzufügen oder Bearbeiten neuer individueller Sensorkonfigurationen.

Usability Aus den Erfahrungen des Praxisbetriebes gab es vereinzelte Rückmeldungen über Bugs und aufgetretene Probleme, aber auch Verbesserungsvorschläge bezüglich der Benutzbarkeit des Systems im Allgemeinen. Die Resonanz war dabei im Wesentlichen positiv, wies aber nichtsdestotrotz vereinzelt auch auf noch ungelöste Schwierigkeiten hin. Diese wurden teilweise in das angestrebte Release 0.2.0 eingearbeitet. Insbesondere erwähnenswert ist an dieser Stelle, dass pro etablierter PHP-Session standardmäßig nicht mehrere Requests parallel abgearbeitet werden können. Dies fällt im Regelbetrieb nicht

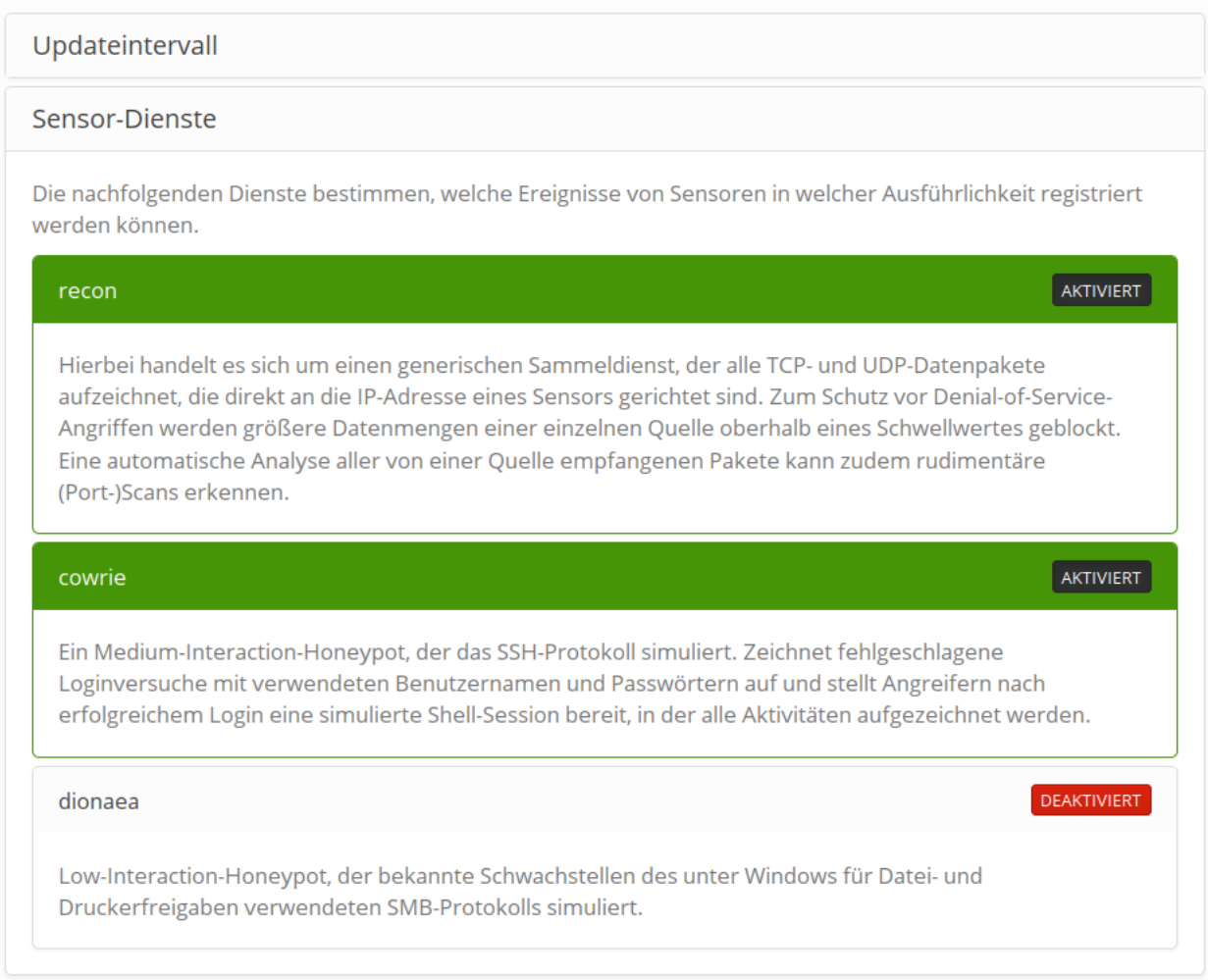

Abbildung 1: Sensorkonfiguration

auf, da einzelne Anfragen binnen sehr kurzer Zeit bearbeitet werden. Ein Sonderfall ist allerdings das Beziehen einer größeren Datenmenge vom Server, wie beispielsweise der Download einer Sensor-Firmware für das anschließende Flashen eines Installationsmediums. Hier war es bisher der Fall, dass bis zum Abschluss des Downloads keine zusätzlichen Verbindungen zum Server blockiert wurden und somit eine Bedienung der Webanwendung während des Downloads nicht möglich war. Diese Limitierung wurde nun durch geschicktes Verwalten der Session-Ressource aufgehoben.

Das vorübergehende Highlighting neu eintreffender Ereignisse war seit der Migration auf die lazy-loading unterstützende *Datagrid*-Ereignisübersicht nicht mehr gegeben und wurde während dieser Projektphase neu implementiert. Eine Übernahme des alten Verfahrens, das noch auf *DataTables*-Schnittstellen basierte, war aufgrund des Architekturunterschiedes beider Lösungen nicht möglich. Da die Verwaltung der Ereignistabelle nun serverseitig stattfindet, mussten jegliche Rahmenbedingungen wie die Filter- und Sortieranforderungen beim Abrufen neuer Ereignisse vom Server berücksichtigt werden. Ein ähnliches Verfahren wird nun auch genutzt, um die graphischen Dashboard-Übersichten live und unter Berücksichtigung der gewählten Filtereinstellungen zu aktualisieren.

Deployment-Prozess Bei der Vorbereitung des 0.2.0-Releases und den damit verbundenen Tests hat sich herausgestellt, dass die automatische Aktualisierung bestehender Installationen unter bestimmten Randbedingungen fehlschlägt. Aus diesem Grund wurde der Update-Worker aktualisiert und ein Bootskript zum Docker-Container hinzugefügt, das diese potentiell bereits vor dem assistierten Update-Vorgang auftretenden Probleme automatisch bereinigt und beispielsweise die Berechtigungen diverser Verzeichnisse zurücksetzt. Im Falle von HoneySens-Installationen mit selbstsignierten TLS-Zertifikaten befand sich das Zitat zudem innerhalb des Docker-Anwendungscontainers und musste zunächst händisch einmalig gesichert und in den Datencontainer übertragen werden. Für zukünftige Installationen wurde die Position des Zertifikates dauerhaft in den Datencontainer verschoben, so dass auch solche Installationen ohne Verlust des für die Server-Sensor-Kommunikation essentiellen Zertifikates aktualisiert werden können.

Sonstiges Es wurden im Rahmen einer Code-Bereinigung weiterhin einige inzwischen ungenutzte Views, Komponenten, externe Abhängigkeiten und Funktionen entfernt und in diesem Zusammenhang auch die Unterstützung für unverschlüsseltes HTTP für die Server-Sensor-Kommunikation sowohl aus der Sensor-Firmware als auch aus der Serversoftware gestrichen.

Weiterhin wurde der Testbetrieb an der TU Dresden ausgeweitet. An diesem nimmt jetzt auch die Fakultät Psychologie mit zwei Netzen teil.

3 Ausblick

In unmittelbarer Zukunft steht die schnelle Ausweitung der Teststellungen im Mittelpunkt. Insbesondere innerhalb des TUD-Netzes sind dafür bereits alle Vorbereitungen abgeschlossen und es ist nur noch die Schulung der Mitarbeiter und die Installation der Sensoren vorzunehmen. In diesem Zusammenhang ist auch die Erstellung und Erweiterung der Benutzerdokumentation von großer Bedeutung. Hier ist auch seitens des Ministeriums für die Administratoren des SVN eine umfangreiche, leicht verständliche Anleitung gewünscht, damit die Verwaltung des Systems in Zukunft bei Bedarf auch ohne externen Dienstleister erfolgen kann.

Weitere Schritte umfassen dann die Integration der noch offenen Punkte des Penetrationstests und die Realisierung eines Signatursystems für Sensorfirmware, um im Falle eines erfolgreichen Angriffes auf den Server zu verhindern, dass der Eindringling auch Zugriff auf alle Sensoren bekommt.

4 Anhang

Diese Auflistung soll die zuvor beschriebenen Prozesse und Veränderungen anhand der im Laufe des Forschungsprojektes in der Versionsverwaltung hinterlegten Kommentare für jede neue Softwarerevision dokumentieren.

Revision	Änderungen
373	- Support for the usage of unencrypted HTTP removed from the API, the frontend and the sensor firmware - Fixed a bug that prevented the doctrine-clip.php script startup - Removed some old leftover template files
374	- Sensor global configuration frontend view rewritten and incorporated into the sensor module - Event filter list sorting can now be toggled between inactive/asc/desc - Some transition region related code snippets cleaned up (no longer needed) - Sensors and event filters now use a PageableCollection for proper cooperation with the Backgrid framework - Firmware Debian package bumped to version 0.2.0 - Fixed a bug that prevented the proper download of sensor config archives - Fixed a bug that prevented proper incremental updates trough the API state resource - Fixed a rare race condition that caused the frontend validation engine to crash because jquery isn't loaded in time
375	- Individual sensor configuration views added - Fixed a bug that prevented observers and managers to filter data on the dashboard and to view past sensor status reports
376	- Kippo SSH honeypot replaced with cowrie (an updated fork) - Managers aren't allowed to change the system-wide sensor configuration anymore - Fixed a few bugs that prevented submitted events from being accepted due to too strict validation rules
377	- Certificate CN detection is now more resistant to failures - Removed some unused JavaScript template dependencies which were causing errors when minified
378	Chart.js updated to version 2.2.1 to fix a bug in combination with require.js minification for the release build
380	Added python-gmpy2 as dependency for the BBB platform. This speeds up the cowrie honeypot considerably.

382	Release of HoneySens 0.2.0 - The dashboard global status data now properly updates incrementally over time - New events that arrive while the event list is displayed are now highlighted for a short time. Filter and order settings are respected as well. - Installer: Improved the certificate common name detection - The HTTPS certificate is now stored in the HoneySens data directory so that it survives docker container updates (relevant for self-signed certs) - Added a docker container boot script that fixes permissions on some directories that caused problems when updating from older versions - Fixed a bug that caused the download of firmware images to block all other requests to the web app until it is finished - Fixed an inconsistency that prevented sensor names with special characters (_-.) on the API, but not the browser side