

Technische Universität Dresden

Fakultät Informatik

Professur für Datenschutz und Datensicherheit

Projekt HoneySens

7. Zwischenbericht

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

Stand: 14.11.2016

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.

Nachdem in der vorangegangenen Projektphase die Version 0.2.0 mit neuen Features und einer großen Zahl entfernter Altlasten veröffentlicht wurde, stand im Rahmen des jetzigen Zeitraumes das Erstellen einer umfangreichen Benutzerdokumentation zur Ergänzung des Probeeinsatzes im Vordergrund. Diese wurde außerdem aus organisatorischen Gründen zur Ausweitung der Testumgebungen benötigt. Zusätzlich wurde die neue Version in den Testumgebungen installiert und auf die im Betrieb gesammelten Erfahrungen und Bugs zeitnah reagiert. Zuletzt wurden eine Reihe von schon seit einiger Zeit existierenden Kompatibilitätsproblemen mit dem *Microsoft Internet Explorer* korrigiert, die ein vernünftiges Arbeiten mit diesem Browser unmöglich machen. Da dieser insbesondere im Sächsischen Verwaltungsnetzwerk in vielen Teilnetzen als einziger Browser vorgeschrieben ist, war dieser Schritt von großer Bedeutung.

Wie auch schon im vorherigen Projektabschnitt war auch in dieser Phase das Ausweiten der Testumgebung innerhalb des Netzwerkes der TU Dresden ein fester Bestandteil. Während weitere Teilnehmer für den Testeinsatz durch entsprechende Vorträge erschlossen wurden, erforderte auch die Wartung des bestehenden Netzes etwas Aufmerksamkeit. So meldeten sich Benutzer bezüglich defekter Geräte, Fragen zur Konfiguration und Bedienung und auch bezüglich gefundener Bugs.

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 18. September 2016 bis zum 14. November 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.

Benutzerdokumentation Ein Großteil der Zeit wurde in die Erstellung einer ausführlichen Benutzerdokumentation investiert. Diese beschreibt für neue Nutzer knapp die Kernideen und die Funktionsweise des Systems, um anschließend detailliert auf die Bedienung der Benutzeroberfläche einzugehen. Dabei werden alle für den täglichen Betrieb relevanten Aspekte der Software angesprochen. Da aus der Erfahrung mit dem Testsystem heraus bereits bekannt war, dass einige Arbeitsabläufe wie beispielsweise das Hinzufügen neuer Sensoren oder das Anlegen von Filterregeln häufig durchgeführt werden müssen und immer wieder zu Fragen führen, wurde das Handbuch nach der ausführlichen Erklärung aller einzelnen Programmbestandteile und Optionen um ein Kapitel *HowTos* erweitert, das solche typischen Arbeitsabläufe sehr bildhaft beschreibt (und dabei immer wieder auf die zugehörigen Abschnitte in der Referenzdokumentation verweist).

Der aktuelle Stand des Benutzerhandbuchs, der abgesehen von noch notwendigen Korrekturen inhaltlich dem vorläufigen Endstand für diese Version entspricht, ist an dieses Dokument angehängt. Bei zukünftigen Weiterentwicklungen wird die kontinuierliche Aktualisierung der Dokumentation berücksichtigt.

Kompatibilität Internet Explorer Im Rahmen der innerhalb des Sächsischen Verwaltungsnetzwerkes installierten Testinstallation hatten sich Probleme bei der Bedienung der Webanwendung mit dem Browser *Microsoft Internet Explorer* herausgestellt. So wurden Teile der Anwendung erst nach mehrmaligem Aktualisieren der gesamten Website sichtbar und auch bei der Bedienung traten vereinzelt Fehler auf. Diesen Problemen wurde in dieser Projektphase nachgegangen. Es hat sich dabei herausgestellt, dass dieser Browser die Ergebnisse von AJAX-Requests in einem Cache vorhält, was ausgesprochen untypisch ist¹. Da alle Bestandteile der Webanwendung im Hintergrund ausschließlich über AJAX-Requests kommunizieren, führte dieses Verhalten zu unerwarteten Problemen in allen Bereichen der Anwendung. Request Caching ist jetzt über alle Browser hinweg deaktiviert, wodurch die Probleme behoben wurden und die Anwendung jetzt wieder browserübergreifend nutzbar ist.

Sonstiges Es wurden weiterhin einige kleinere Probleme behoben, die im Rahmen des andauernden Testbetriebes entstanden sind. Hierzu zählt auch ein potentiell Datenschutzproblem, bei dem Daten einer alten Session nach einem Logout noch immer im Browser verfügbar waren. Weiterhin wurden Codeabschnitte bereinigt, die noch exklusiv für die inzwischen nicht mehr zur Verfügung stehenden reinen HTTP-Kommunikation mit dem Server genutzt wurden.

Weiterhin wurde der Testbetrieb an der TU Dresden ausgeweitet. An diesem nimmt in naher Zukunft das *Universitätsklinikum Carl Gustav Carus* teil. Nach einem Vortrag und Absprache mit der *Leitstelle für*

¹<http://www.dashbay.com/2011/05/internet-explorer-caches-ajax/>

Informationstechnologie der sächsischen Justiz wurde das Aufstellen von Sensoren auch in diesem Netz in Aussicht gestellt, das Teil des SVN ist.

3 Ausblick

Als unmittelbar nächste Schritte sind weitere Dokumentationen notwendig. Darunter die Erstellung von Dokumentation zur Verwaltung und Administration des Servers, was insbesondere auch den Umgang mit Docker-Containern betrifft. Außerdem soll das Sicherheitskonzept zum Härten des HoneySens-Servers dokumentiert und einige Zuarbeiten für die interne organisatorische Betriebsdokumentation geleistet werden.

Weiterhin steht die schnelle Ausweitung der Teststellungen im Mittelpunkt. Es existieren noch eine Reihe weiterer Fakultäten, die Interesse an einer Teilnahme am Testbetrieb geäußert hatten und diesbezüglich nun kontaktiert werden sollen.

Längerfristig sind dann auch wieder Erweiterungen an der Codebasis geplant: Zunächst ist das nach erneuter Re-Priorisierung eine Überarbeitung der E-Mail-Notifikation. Diese ist im Moment sehr starr gestaltet und lässt abgesehen von der Definition der gewünschten Adressaten nur wenig Spielraum bei der Definition der Art der zu sendenden Nachrichten. Es gilt daher, ein Konzept zu erarbeiten, wie genauer festgelegt werden kann, worüber die Empfänger informiert werden sollen und dieses dann auch zu implementieren. Zukünftig steht dann ebenso die Realisierung eines Signatursystems für Sensorfirmware auf dem Plan, um im Falle eines erfolgreichen Angriffes auf den Server zu verhindern, dass Eindringlinge auch Zugriff auf alle Sensoren erhalten.

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
383	- Leftover HTTP sensor code removed
384	- Server version bump to 0.2.1
387	- Fixed a bug that caused cached model data to carry over to new sessions after logout
389	- Compatibility fixes for Internet Explorer: Aggressive AJAX request caching is now disabled

Benutzerhandbuch

für Version 0.2.1

Stand: November 2016



geschrieben von
Pascal Brückner

Inhaltsverzeichnis

I	Einleitung	1
II	Funktionsprinzip	2
1	Server	4
2	Sensoren	5
III	Betrieb	6
3	Login	7
4	Benutzerinterface	8
5	Dashboard	9
6	Ereignisliste	9
7	Ereignis-Filter	11
8	Sensoren	13
9	Benutzer- und Gruppenverwaltung	20
10	Systemeinstellungen	22
IV	HowTos	25
11	Einrichten eines neuen Sensors	25
12	Ereignisfilter anlegen	29

Teil I

Einleitung

Der *Internet Crime Report* des *Internet Crime Complaint Centers*, welches weltweit Daten über die durch Cyberkriminalität verursachten Schäden zu erheben versucht, verzeichnete für das Jahr 2013 mehr als 260.000 Vorfälle und einen Gesamtverlust von über 780 Millionen Dollar. Für den gleichen Zeitraum nennt der *Norton Cybercrime Report* hingegen angerichtete Schäden im Wert von 113 Milliarden Dollar und 378 Millionen direkt Betroffene. Der Bericht besagt weiterhin, dass 41 % aller sich im Netz bewegendes Erwachsenen bereits einmal Opfer von Malware, Phishing oder vergleichbaren Angriffen geworden sind. Es wird schnell deutlich, dass es aufgrund des schieren Ausmaßes dieser Form von Kriminalität sehr schwierig geworden ist, darüber allgemeingültige Aussagen zu treffen; insbesondere unter Berücksichtigung der Tatsache, dass die Entwicklung von auf spezifische IT-Infrastrukturen zugeschnittener Malware in den letzten Jahren zugenommen hat, wie es das Beispiel *Stuxnet* bereits im Jahr 2010 demonstrierte. Im *Verizon Data Breach Investigations Report* für das Jahr 2014 wurde weiterhin deutlich, dass Angriffe der Kategorie *Insider Misuse* und *Cyber-espionage* konstante Bedrohungen darstellen. Ein Angreifer, der physischen Zugang zum internen Netzwerk besitzt oder womöglich über Techniken des *Social Engineerings* privilegierten Zugang erhalten hat, aber auch Mitarbeiter, die ihre Privilegien aus welchen Gründen auch immer überschreiten, stellen somit eine ernst zu nehmende Gefahr für die Integrität einer jeden komplexen IT-Infrastruktur dar. Die möglichen Wege der Infektion von Produkivsystemen mit Malware sind durch in manchen Unternehmen zulässigen Prinzipien wie *Bring your own device* inzwischen vielfältiger Natur, ebenso kann aber auch schon der Besuch einer attackierenden Website oder das unbedachte Öffnen eines scheinbar harmlosen E-Mail-Anhangs einen Einbruch in das Netzwerk nach sich ziehen. All diesen Fällen ist gemein, dass die das interne Netzwerk vom Internet trennenden Schutzmaßnahmen versagt haben oder aber bedingt durch die Art des Angriffes gar keine Wirkung zeigen konnten. *Honeypots* stellen ein erprobtes Mittel dar, das in Zusammenarbeit mit anderen Technologien wie *Intrusion Detection Systems* und *Anti-Viren-Software* die Sicherheit innerhalb eines Netzwerks verbessern und auch neue, während eines Vorfalls noch unbekannte Angriffsmuster erkennen kann.

Die HoneySens-Plattform ist ein auf der Idee von Honeypots basierendes Werkzeug zur Absicherung von IT-Landschaften, das speziell auf derartige Angriffe *aus dem Inneren* hin optimiert ist und dabei helfen soll, die Bedrohungslage innerhalb einer komplexen Netzwerkarchitektur in kurzer Zeit und mit nur wenigen zusätzlichen Ressourcen zu analysieren. Bei der Entwicklung wurde Wert darauf gelegt, dass das System auch für Administratoren ohne tieferen Hintergrund im Bereich der IT-Sicherheit leicht verständlich ist und sich in eine bestehende Netzinfrastruktur möglichst transparent integrieren lässt, ohne die bereits vorhandenen produktiven Komponenten zu beeinträchtigen oder umfangreiche Maßnahmen zur Konfiguration von Switches, Firewalls usw. vorauszusetzen.

Dieses Dokument gibt einen Überblick über die Funktionsweise und Bedienung des Systems und seiner Komponenten. Im zweiten Abschnitt wird dafür zunächst ein knapper Überblick über die Gesamtarchitektur der aus Hard- und Softwarekomponenten bestehenden Lösung und deren Zusammenspiel gegeben¹. Kapitel drei erklärt anschließend detailliert, wie sich HoneySens im praktischen Betrieb nutzen lässt.

¹Für zusätzliche Informationen siehe auch *Brückner, Pascal: Realisierung eines Honeypot-Netztes (2014)*

Teil II

Funktionsprinzip

Die Grundidee beim Einsatz von HoneySens ist, sogenannte *Sensoren* innerhalb der einzelnen Teilnetze einer IT-Landschaft zu platzieren, die verdächtigen Datenverkehr im Netzwerk aufzeichnen und an eine zentrale Serverinstanz zur Auswertung weiterzuleiten. Als Sensoren kommen Kleinstrechner vom Typ *BeagleBone Black* oder *BeagleBone Green* zum Einsatz, die sich durch einen geringen Stromverbrauch, geringe Wärmeentwicklung und einen niedrigen Anschaffungspreis auszeichnen. Die Serverplattform wird in Form eines *Docker-Containers*² bereitgestellt und kann mit jeder modernen Linux-Distribution ohne die aufwändige Installation zusätzlicher Abhängigkeiten schnell genutzt werden. Damit die Sensoren mit dem Server kommunizieren können, müssen insbesondere die in produktiven Netzen zumeist restriktiv konfigurierten Firewalls und eventuell vorhandene Proxy-Server passiert werden. Mit dem Hintergrund der möglichst transparenten Integration der HoneySens-Komponenten in ein bestehendes Netz wurde außerdem die Errichtung eines weiteren separaten Honeypot-Subnetzes, in das verdächtige Anfragen umgeleitet werden, ausgeschlossen: Dieser Schritt wäre mit zusätzlichem organisatorischem Aufwand und dem Einsatz von *Intrusion Detection Systemen (IDS)* zur Detektion verdächtiger Datenströme verbunden. Typischerweise sind bestehende Schutzsysteme wie Firewalls so konfiguriert, dass Systeme aus den Subnetzen heraus mit Rechnern im Internet kommunizieren können. Das Web-Protokoll *HTTP*, bzw. dessen verschlüsselte Variante *HTTPS* ist deshalb normalerweise für ausgehende Verbindungen freigegeben. Hinzu kommt der Umstand, dass es die restriktiven Firewalls typischerweise nicht gestatten, dass der Server seine Sensoren von außerhalb des Teilnetzes aus direkt kontaktieren kann.

Den genannten Umständen wird die HoneySens-Architektur gerecht, indem die Kommunikation zwischen den Teilnehmern ausschließlich über *HTTPS* erfolgt und der Kommunikationskanal zwischen Sensoren und Managementserver *gerichtet* ist, sodass lediglich sichergestellt werden muss, dass alle Sensoren von ihrer Schnittstelle aus eine neue Verbindung zum Server auf dem TCP-Port 445 aufbauen können, was in Unternehmensnetzen häufig bereits der Fall ist. Eine gesonderte Rekonfiguration der Firewalls ist folglich oft gar nicht nötig. In Abbildung 1 ist eine HoneySens-Installation mit mehreren Sensoren in ihren jeweiligen Teilnetzen und einem von allen Punkten aus zentral erreichbaren Server dargestellt. Die Sensoren sind weitestgehend unabhängig von externer Infrastruktur und beziehen beispielsweise auch ihre Systemzeit direkt über *HTTPS* vom Server. Eine Kommunikation zwischen Server und Sensor über andere Protokolle ist somit nicht nötig. Für den Fall, dass ein Zugang zum HoneySens-Server ausschließlich über *HTTP-Proxy-Server* möglich ist, können Sensoren zur Nutzung dieser nach entsprechender Konfiguration angewiesen werden.

Da aus den genannten Gründen auf ein separates Honeypot-Teilnetzwerk, wie es teilweise in anderen Honeypot-Installationen zum Einsatz kommt, verzichtet wird, werden die Honeypot-Dienste in einer HoneySens-Installation *lokal* auf den Sensoren betrieben. Die Honeypots sind für Angreifer somit bezüglich ihrer Latenz nicht von den umgebenden Rechnern zu unterscheiden, was in manchen auf umfangreichen Routing-Strecken oder VPN basierenden Lösungen nicht der Fall ist. Um das die Sensoren umgebende Netz nicht zu gefährden, kommen weiterhin ausschließlich sog. *Low Interactive Honeypots* zum Einsatz, die aus Angreifersicht nur eine verminderte Interaktion mit dem Gerät ermöglichen und somit von einem intelligenten Angreifer verhältnismäßig schnell als Falle enttarnt werden können - nicht

²Siehe auch <http://docker.io>

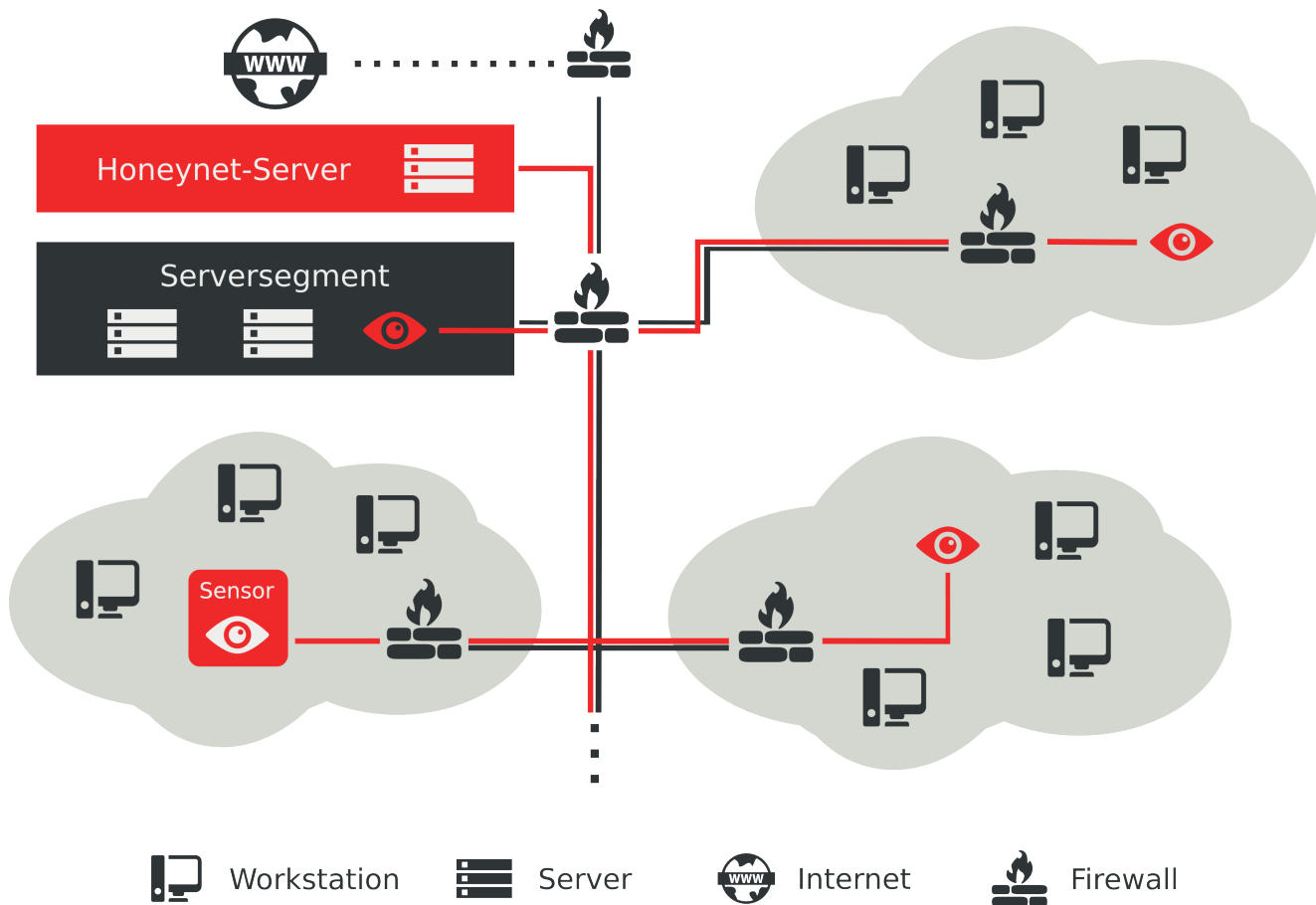


Abbildung 1: Topologie einer HoneySens-Installation

jedoch, bevor bereits eine Meldung über den Vorfall entstanden ist. Der Vorteil solcher Honeypots liegt jedoch zugleich auch im deutlich geringeren Risiko, dass der Angreifer mit Hilfe des Honeypots seine Rechte im Netzwerk ausweiten kann. Derartige Honeypots sind so entworfen, dass Eindringlinge lediglich mit simulierten Diensten und Protokollen interagieren und eine Übernahme durch externe Angreifer nicht möglich ist, da Schwachstellen der lediglich vorgetäuschten Protokolle nicht ausgenutzt werden können. Weil die Honeypots dezentral verteilt werden, stellt HoneySens neben Möglichkeiten zur Auswertung der aufgezeichneten Ereignisse auch umfangreiche Funktionen zur Verwaltung der Sensorflotte zur Verfügung. Diese umfassen beispielsweise auch automatische Firmware-Aktualisierungen für alle angeschlossenen Sensoren, um den administrativen Aufwand möglichst gering zu halten.

Die Kommunikation zwischen Sensoren und Server erfolgt verschlüsselt über TLS. Mit Hilfe von sogenanntem *Certificate Pinning* wird außerdem sichergestellt, dass die Sensoren mit dem richtigen Server verbunden sind und bei *Man-in-the-Middle*-Angriffen (MITM) Vertraulichkeit sichergestellt ist. Analog kommuniziert der Server ausschließlich mit Sensoren, die ein gültiges, speziell für sie ausgestelltes Zertifikat vorweisen können und ihre Nachrichten mit diesem signieren.

In den nachfolgenden Abschnitten wird detailliert auf die Funktionsweise des Servers und der Sensoren eingegangen.

1 Server

Die zentrale Anlaufstelle für alle Sensoren des HoneySens-Systems stellt der an einer von allen Knoten erreichbaren Stelle platzierte Server dar. Um die Hardware- und Integrationskosten des Gesamtsystems möglichst gering zu halten, werden alle Verwaltungs- und Analysefunktionen gemeinsam von diesem System übernommen. Dies umfasst unter anderem die folgenden Funktionen:

- Speichern und Auflisten aller an den Sensoren aufgetretenen Ereignisse und die Möglichkeit, relevante Details über diese (wie beispielsweise fehlgeschlagene Login-Versuche an einem SSH-Honeypot) abzurufen. Funktionen zum Sortieren dieser Liste gehören zusätzlich zum geplanten Funktionsumfang.
- Bereitstellung einer Übersicht über alle registrierten Sensoren und deren aktuellen Status (online/offline/im Updateprozess etc.)
- Speichern und Auflisten von **Statusinformationen** der Sensoren, wie die aktuell verwendete Software-Revision oder die momentane Auslastung.
- Bereitstellung von Infrastruktur zum **Aktualisieren der Sensorsoftware**: Verwaltung und automatisierte Verteilung sowie Installation von Firmware-Images.
- **Server für die Zeitsynchronisation**: Da HTTP(S) als alleiniger Kommunikationskanal zwischen Sensoren und Server genutzt werden soll, erfolgt über diesen auch die Synchronisation der Sensor-Systemzeit
- Sicherstellen der **verschlüsselten Kommunikation zwischen allen Entitäten**: Sowohl die Kommunikation zwischen Benutzer und Server als auch zwischen Sensoren und Server erfolgt verschlüsselt. Hierfür tritt der Server als *Certificate Authority* (CA) auf und stellt Zertifikate für die Sensoren aus, mit denen diese alle Anfragen signieren.
- **Infrastruktur zur benutzerfreundlichen Sensorverwaltung**: Dies umfasst primär das Installieren neuer Sensoren, speziell die Registrierung derer beim Server, das Anbieten von Firmware-Installationsmedien zum Download und die Integration der spezifischen Konfigurationsoptionen für jeden Sensor. Beim Entfernen von Sensoren auf dem Server ist zudem sichergestellt, dass jegliche mit dem Gerät verbundenen Einstellungen und Ereignisse ebenfalls vernichtet werden.
- Verwaltung verschiedener Firmware-Revisionen sowie die Möglichkeit, einzelne Sensoren selektiv mit bestimmter Firmware zu versorgen, um beispielsweise die Funktionalität einer neuen Softwareversion zuerst mit einigen wenigen Sensoren zu testen.
- **Notifikation via E-Mail**: Um interessierte Personen in regelmäßigen Intervallen über die Arbeit des Systems informieren zu können, kann der Server wöchentlich Zusammenfassungen über die aufgetretenen Ereignisse und den Status der Sensoren (online/offline) via E-Mail verschicken. Zusätzlich kann eine *sofortige Notifikation* von zuständigen Stellen über E-Mail im Falle eines kritischen aufgetretenen Ereignisses erfolgen.
- Graphische Aufbereitung und Anzeige von Statistiken über die gesammelte Datenmenge.
- **Mandantenfähigkeit: Unterstützung mehrerer Benutzer und Gruppen** mit unterschiedlichen Rechten: Eine Benutzerverwaltung, in der verschiedene „Rollen“ für die Accounts vergeben werden, stellt sicher, dass einem angemeldeten Benutzer nur die ihm erlaubten Optionen präsentiert werden. Durch die Benutzerverwaltung wird zudem sichergestellt, dass nicht-autorisierten Personen der Zugang zum System verwehrt bleibt.
- **Whitelist-Filter**: Regelmäßig wiederkehrende, aber als harmlos einzustufende Ereignisse können mit Hilfe von frei definierbaren Filterregeln und einer *Whitelist* gezielt verworfen werden.

Unter Berücksichtigung der im vorigen Kapitel besprochenen Voraussetzung, die Kommunikation zwischen Sensor und Server ausschließlich auf das HTTPS-Protokoll (TCP/443) zu beschränken, und aufgrund der Möglichkeit, dass Firewalls mit Hilfe von *Deep Packet Inspection* Pakete auf diesen Ports, die sich nicht an die HTTP(S)-Protokolldefinition halten, einfach verwerfen, wurden für die Serverschnittstelle ein webbasiertes Front- und Backend konzipiert. Kernkomponente der Architektur ist eine *API*³, über die alle Sensoren mit dem Server Daten austauschen. Da eine webbasierte API aus diesem Grund ohnehin über das Netzwerk erreichbar sein muss, können Clients als „Honeynet-Frontend“ in beliebigen Programmiersprachen und für beliebige Architekturen entwickelt werden. Somit kann diese Schnittstelle neben der mit dem Browser nutzbaren Web-Schnittstelle auch aus Skripten heraus genutzt werden.

Der Server ist somit zentraler Endpunkt für die Kommunikation sowohl mit den Sensoren als auch regulären Benutzern. Beide Fälle unterscheiden sich aber insofern, als die Gültigkeit der Nachrichten von Sensoren mit Hilfe einer *Public Key Infrastructure* (PKI) gewährleistet wird, während sich Benutzer mit einer Kombination aus Benutzername und Passwort beim System anmelden. Sensoren weisen zudem ein anderes Kommunikationsverhalten auf: Neben Nachrichten mit gebündelten Informationen über die aufgetretenen Ereignisse kontaktieren die Geräte in regelmäßigen, frei definierbaren Zeitabständen den Server (sog. *Polling*), um ihre aktuelle Konfiguration abzufragen, Statusdaten wie die aktuelle Systemauslastung bereitzustellen und potentiell anstehende Firmwareaktualisierungen vorzubereiten.

2 Sensoren

Hauptaufgabe der sogenannten Sensoren im HoneySens-System ist das Sammeln von Informationen über verdächtige Ereignisse innerhalb des Netzwerks. Dabei kann es sich sowohl um sich selbstständig ausbreitende Schadsoftware („Malware“), aktive menschliche Angreifer oder auch Innentäter handeln. Die Sensoren operieren dabei grundsätzlich dem *Honeypot*-Prinzip: Sie sind passive Teilnehmer im Netzwerk mit eigener IP-Adresse, initiieren also abgesehen von ihrer Verbindung zum HoneySens-Server sonst keine Verbindungen zu anderen Rechnern und strahlen auch keine aktiven Broadcast- oder Multicast-Pakete aus. Wenn nun von einem Sensor Datenpakete empfangen werden, die gezielt an seine IP-Adresse gerichtet sind, handelt es sich zwingend um ein verdächtiges Ereignis, da der Sensor allen anderen Systemen im Netzwerk unbekannt sein sollte. Im Normalfall sollte ein Sensor daher niemals direkt an ihn gerichtete Pakete erhalten. Von diesem Prinzip abweichende Pakete werden zur späteren Auswertung protokolliert. Um noch zusätzliche Informationen über potentielle Angreifer gewinnen zu können, können auf den Sensoren zusätzlich noch Honeypot-Dienste aktiviert werden. Diese Low-Interaction-Honeypots simulieren verschiedene Protokolle, um Angreifern mehr Interaktivität mit dem Sensor zu ermöglichen und letztendlich etwas über die Motive und Vorgehensweise potentieller Eindringlinge zu erfahren.

HoneySens gibt grundsätzlich keine Beschränkungen bezüglich der maximal einsetzbaren Anzahl an Sensoren vor. Es ist sinnvoll, mindestens einen Sensor in jedem Netzsegment zu platzieren, das auf verdächtige Ereignisse hin überwacht werden soll. Wenn jedoch mehrere IP-Adressen zugleich mit Sensoren belegt werden sollen, was beispielsweise die Wahrscheinlichkeit erhöht, auf verdächtige Malware aufmerksam zu werden, sind derzeit auch mehrere Sensoren erforderlich.

³Application Programming Interface

Teil III

Betrieb

Dieser Teil des Handbuchs erklärt, wie Benutzer die durch HoneySens bereitgestellte Funktionalität nutzen, gesammelte Daten auswerten und das System verwalten können. All diese Funktionen werden vom Server in Form eines Web-Interfaces angeboten und können mit jedem modernen Browser genutzt werden. Jeder Benutzer erhält hierfür von einem Administrator des Systems einen Account mit zugehörigem Login-Namen und Passwort. HoneySens ist außerdem mandantenfähig und stellt verschiedene, feste Benutzerrollen zur Verfügung. Somit ist sichergestellt, dass jeder Nutzer ausschließlich die von ihm verwalteten Sensorgruppen einsehen kann und - falls nötig - mit eingeschränkten Rechten zur Änderung von deren Konfiguration ausgestattet ist.

Jeder Account ist einer der folgenden drei Rollen zugewiesen, die maßgeblich bestimmen, welche Rechte der entsprechende Benutzer im System besitzt.

Beobachter : Die schwächste Benutzerrolle erlaubt das Einsehen aller registrierten Sensoren und deren Status, aller von diesen gemeldeten Ereignisse und zugehörigen Details, die auf dem Server hinterlegten Firmware-Revisionen und das Dashboard mit einer graphischen Aufbereitung der gesammelten Ereignisdaten. Der Einfluss eines Beobachters beschränkt sich auf das reine Einsehen der Daten. Eine Veränderung der Sensoreinstellungen oder das Entfernen von Ereignissen ist nicht möglich. Beobachter besitzen somit ausschließlich die Möglichkeit zur Auswertung der gesammelten Daten und ist beispielsweise für externe Dritte gedacht, die lediglich einen statistischen Einblick in eine HoneySens-Installation benötigen, aber keine Verwaltungsaufgaben wahrnehmen, aber keine Verwaltungsaufgaben wahrnehmen.

Manager : Benutzer mit dieser Rolle besitzen ebenfalls vollen Einblick in die Liste der Sensoren und registrierten Ereignisse aus allen Gruppen, denen der Account angehört. Zusätzlich sind jedoch auch Modifikationen möglich: Sensoren können hinzugefügt, konfiguriert und entfernt werden. Ereignisse können entfernt, kommentiert oder ihr Status angepasst werden. Zusätzlich können Manager sogenannte *Filter* anlegen, die in Form einer Whitelist gezielt harmlose, wiederkehrende Vorfälle als unkritisch klassifizieren und nicht mehr als gesondertes Ereignis melden. Zuletzt ist es Managern gestattet, die globale Sensorkonfiguration einzusehen, aber auch individuelle Einstellungen für einzelne Sensoren gesondert zu tätigen. So können beispielsweise gezielt HoneyPot-Dienste auf einzelnen Sensoren aktiviert oder deaktiviert werden.

Administrator : Nutzer mit administrativen Rechten erhalten zusätzlich zu allen zuvor genannten Rechten einen Einblick in alle vorhandenen Sensorgruppen, die zugehörigen Sensoren und alle im System registrierten Ereignisse und Ereignisfilter. Zusätzlich steht Administratoren die Benutzer- und Gruppenverwaltung vollumfänglich zur Verfügung und sie haben die Möglichkeit, die globale Sensorkonfiguration anzupassen. Diese ist für alle Sensoren gültig, für die keine individuellen Einstellungen hinterlegt wurden. Zuletzt besitzen Administratoren auch Zugriff auf die Firmware-Verwaltung mit der Möglichkeit, neue Firmware-Revisionen auf dem Server zu registrieren und die global gültige Standard-Revision festzulegen.

Die Auswirkungen der Benutzerrolle auf die jeweils zur Verfügung stehenden Operationen im Webinterface wird in den folgenden Abschnitten jeweils gesondert eingegangen.

An dieser Stelle soll noch kurz das Konzept der **Gruppe** erklärt werden. Im Rahmen der Mandantenfähigkeit können global beliebig viele Gruppen definiert werden, in die sich alle weiteren Aspekte des Systems eingliedern. Sensoren, Ereignisse, Filterregeln usw. sind immer einer festen Gruppe zugeordnet und können nur von Benutzern eingesehen werden, die ebenfalls Mitglied dieser Gruppe sind. Dieses Konzept zieht sich durch alle Teile der Webanwendung und kann genutzt werden, um einen gemeinsamen HoneySens-Server auch in großen Netzwerken gemeinsam zwischen verschiedenen Abteilungen und mehreren organisatorisch getrennten Benutzergruppen aufteilen zu können. Nutzer, die einer fiktiven Gruppe *A* angehören, aber nicht einer Gruppe *B*, können ausschließlich die zu ihrer eigenen Gruppe *A* gehörigen Datensätze einsehen.

3 Login

Die Anmeldung am System erfolgt, wie auch jegliche weitere Interaktion, über einen Web-Browser. Dafür genügt es, die URL des HoneySens-Servers im Browser aufzurufen:

```
https://<honeysens-server>/
```

Falls auf den Server nur per HTTP statt der verschlüsselten Variante HTTPS zugegriffen wird, erfolgt eine automatische Weiterleitung auf das sichere Protokoll. Eine unverschlüsselte Interaktion über HTTP ist mit dem Server nicht möglich. Nach dem Aufruf der Website präsentiert sich dem Benutzer das nebenstehend abgebildete Login-Formular.

Nach dem Eintragen von Zugangsdaten und einem Klick auf *Anmelden* wird der Benutzer nach einer erfolgreichen serverseitigen Überprüfung auf das Dashboard weitergeleitet. Falls die angegebenen Daten ungültig waren, wird dem Benutzer eine entsprechende Fehlermeldung präsentiert.

Es kann vorkommen, dass nach einer erfolgreichen Authentifizierung der Installations- und Update-Assistent mit der Fehlermeldung „Es ist ein Fehler aufgetreten“ erscheint. Dies ist immer genau dann der Fall, wenn gerade die Server-Software aktualisiert wurde, sich aber noch kein Administrator am System angemeldet hat, um den Aktualisierungs-Assistenten auszuführen. Dieser ist dafür verantwortlich, den vorhandenen Datenbestand für die neue Version zu migrieren. Im Falle einer solchen Meldung gilt es, umgehend einen Administrator zu kontaktieren. Ein Login für reguläre Benutzer ist erst dann wieder möglich, wenn der Aktualisierungsvorgang abgeschlossen wurde.

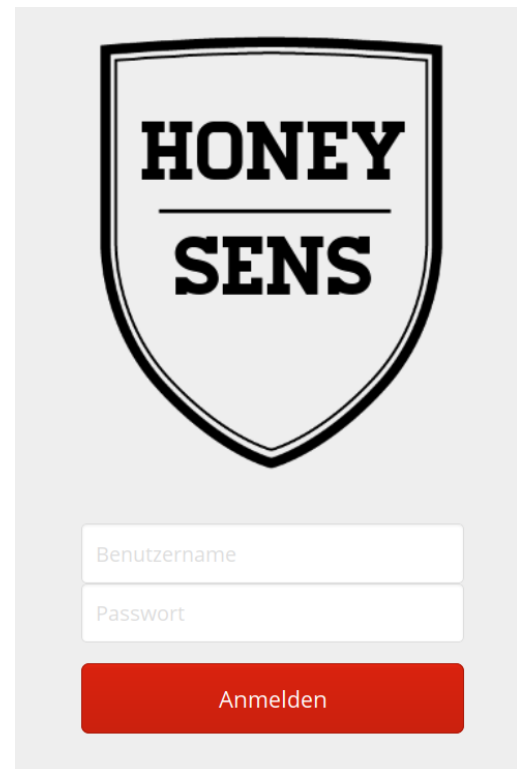


Abbildung 2: Login-Formular

4 Benutzerinterface

Nach der erfolgreichen Anmeldung am System präsentiert das HoneySens-Webinterface das Dashboard. Dabei handelt es sich um eine graphische Übersicht über die im vergangenen Zeitraum aufgezeichneten Ereignisse. Zunächst soll jedoch ein Blick auf die allgemeine Struktur der GUI gegeben werden. Auf die einzelnen Module wird im Anschluss gesondert eingegangen. Abbildung 3 zeigt, wie die GUI in drei große Bereiche aufgeteilt ist:



Abbildung 3: Die Einteilung des Benutzerinterfaces

Header : Der Kopfteil der Seite gibt Aufschluss über die Session des derzeit angemeldeten Benutzers. In der rechten oberen Ecke steht der Benutzername. Direkt daneben befindet sich ein klickbarer „Logout“-Link, der die aktuelle Session beendet und ein Zurückkehren zum Login ermöglicht. Links neben dem Benutzernamen ist weiterhin ein kreisrunder Indikator, der anzeigt, wann alle Daten im Webfrontend im Hintergrund aktualisiert werden. Dieser Wert beträgt standardmäßig 10 Sekunden.

Sidebar : Auf der linken Seite des Fensters können in einem vertikalen Menü die verschiedenen Module der Anwendung aufgerufen werden. Um Platz zu sparen, werden normalerweise lediglich Symbole angezeigt. Das derzeit aktive Modul ist mit einem roten Hintergrund hinterlegt. Um ein anderes Modul aufzurufen, genügt es, mit der Maus über die Sidebar zu fahren und dieses anzuklicken. Die Sidebar wird sich dabei über den Inhaltsbereich hinweg vergrößern und zu jedem Modul ein beschreibendes Label einblenden.

Inhaltsbereich : Den größten Teil der Seite nimmt der sogenannte Inhaltsbereich ein. Hier werden dynamisch die Inhalte des jeweils aktiven Moduls angezeigt, also beispielsweise die Ereignisübersicht oder die Sensorliste.

Wenn das Webinterface auf einem Mobilgerät mit einer geringen Displaygröße aufgerufen wird, passt sich die Darstellung an und verfällt in einen gesonderten Modus, bei dem die Sidebar verschwindet und deren Inhalte in ein Dropdown-Menü integriert, das mit einem Klick oder Tap in den oberen rechten Bereich des Headers sichtbar wird.

5 Dashboard

Das Modul *Übersicht* fasst alle bisher aufgetretenen Ereignisse nach ausgewählten Kriterien zusammen. Zur Definition der Kriterien kann die im oberen Bereich angesiedelte Filterleiste genutzt werden (siehe Abbildung 4). Mit dieser ist eine

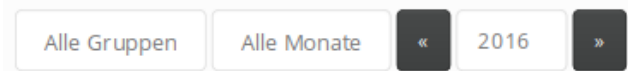


Abbildung 4: Filtermöglichkeiten für das Dashboard

Eingrenzung der dargestellten Daten nach Sensor-Gruppe, Monat oder Jahr möglich. Auch eine Kombination dieser Kriterien ist kein Problem. Eine Reihe von *Widgets* bereiten dann die für den ausgewählten Zeitraum relevanten Daten graphisch auf. Das Widget direkt unter Filterleiste bereitet nach Monaten oder Tagen bereitet die Anzahl der im Zeitraum aufgezeichneten Ereignissen mit Hilfe eines Balkendiagrammes auf, während das Tortendiagramm darunter die im Zeitraum gezählten Ereignisse gemäß ihrer internen Klassifikation aufschlüsselt. So kann der Anteil der kritischen Ereignisse leicht im Verhältnis zur Gesamtzahl betrachtet werden.

6 Ereignisliste

Dieses Modul präsentiert dem Benutzer eine Liste mit allen Ereignissen, die in seinen Gruppen bisher aufgetreten sind. Standardmäßig werden alle Ereignisse aufgelistet, mit den Steuermöglichkeiten auf der Filterleiste über der Tabelle kann die Darstellung aber auch gezielt auf bestimmte Gruppen, Sensoren oder Ereignisklassifikationen reduziert werden. Auch eine Kombination der Filter ist möglich. Die Spalten der Tabelle lassen sich mit einem Klick auf den Tabellenkopf sortieren. So bewirkt beispielsweise ein erster Klick auf *Quelle*, dass diese Spalte alphabetisch aufsteigend sortiert wird. Ein weiterer Klick auf *Quelle* kehrt die Sortierrichtung um. Weiterhin kann zwischen den verschiedenen Seiten der Ereignisliste mit den Buttons unterhalb der Tabelle umgeschaltet werden.

27	28.08.2016 11:08:11	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu			
----	---------------------	-----	--------------------	----------	------------------	-----	--	--	--

Abbildung 5: Ein Ereignis der Ereignisliste

Abbildung 5 zeigt beispielhaft die Darstellung eines einzelnen Vorfalles innerhalb der Liste. Die verschiedenen Spalten haben die folgende Bedeutung:

ID : Die interne Identifikationsnummer für den Vorfall. Hilfreich, um im Gespräch mit anderen Benutzern bestimmte Ereignisse zu referenzieren.

Zeitpunkt : Datum und genaue Uhrzeit, an dem das Ereignis am Sensor aufgezeichnet wurde. Falls der Vorfall mehrere Pakete über einen längeren Zeitraum involviert (beispielsweise im Fall eines Portscans), bezeichnet dies den Zeitpunkt des Eintreffens des ersten Datenpakets.

Sensor : Der Name des Sensors, an dem das Ereignis registriert wurde.

Klassifikation : Die interne Klassifikation des Vorfalls. Diese wird anhand der Ereignisseigenschaften vom Server automatisch vorgenommen: Falls ein HoneyPot-Dienst wie *cowrie* oder *dionaea* den Vorfall aufgezeichnet hat, ist die gesamte Zeile rot hinterlegt und der Vorfall wird als kritisch eingestuft. Reine Verbindungsversuche über TCP oder UDP, wie sie der *recon*-Dienst registriert, sind dagegen wie in der Abbildung nicht gesondert gekennzeichnet, da ein konkretes Gefahrenpotential nicht automatisch angenommen werden kann.

Quelle : Die IP-Adresse, von der dieses Ereignis aus ausgelöst wurde. Dies ist im Normalfall die Absenderadresse der empfangenen Datenpakete. Beachten Sie, dass diese Adresse nicht notwendigerweise auch die des direkten Angreifers sein muss. Technologien wie *NAT*⁴ oder *VPN*⁵ können zur Verschleierung der IP-Adresse eines Angreifers beitragen.

Details : Zusätzliche Informationen zum jeweiligen Ereignis. Die Art dieser Information hängt davon ab, von welchem HoneyPot-Dienst das Ereignis registriert wurde. In der Regel spezifiziert dieses Feld das Ereignis genauer, um eine Einordnung bei einem schnellen Blick auf die Liste zu vereinfachen.

Status : Dieses Feld kann die Werte *Neu*, *In Bearbeitung*, *Erledigt* oder *Ignoriert* annehmen und dient zur Information der Benutzer. Wenn ein Vorfall betrachtet und ausgewertet wurde, kann ein Nutzer mit Manager- oder Administratorrolle den Wert dieses Feldes verändern und zusätzlich noch einen Kommentar hinterlassen. Hierfür genügt es, mit dem Mauszeiger auf das Stift-Symbol zu zeigen. Es öffnet sich anschließend ein kleines Dialogfeld, in dem der Statuswert und optional ein Kommentar gesetzt werden können (siehe Abbildung 6). Mit einem Klick auf den roten Haken-Button werden die Änderungen gespeichert. Die hier eingetragenen Werte haben keinerlei Einfluss auf die Funktionsweise des Systems und dienen lediglich zur Vereinfachung der Übersicht für die Benutzer.

Aktionen : In der letzten Spalte befinden sich zwei Buttons. Der erste mit dem Symbol einer stilisierten Tabelle öffnet einen Dialog, der zusätzliche Details zum Ereignis beinhaltet. Dies umfasst beispielsweise im Falle eines vom *recon*-Dienst aufgezeichneten Vorfalls eine Liste aller vom entfernten Host empfangenen Pakete und deren Inhalte oder durch den SSH-HoneyPot *cowrie* registrierte Authentifizierungsversuche mit zugehörigem Benutzername und Passwort. Der Button mit dem Kreuz-Symbol dient zum unwiderruflichen Entfernen von Ereignissen.

Gemeinsam mit der periodischen Aktualisierung der Datenbasis, angezeigt durch den kreisrunden Indikator im Header, werden auch neu eingetroffene Ereignisse in der Webanwendung angezeigt. Wenn live Vorfälle eintreffen und gerade die Ereignisliste angezeigt wird, findet eine automatische Aktualisierung statt. Neue Ereignisse werden außerdem für kurze Zeit graphisch hervorgehoben.

⁴Network Address Translation

⁵Virtual Private Network



Abbildung 6: Popup zur Bearbeitung des Ereignisstatus

7 Ereignis-Filter

Das Modul *Filter* erlaubt die Definition einer *Whitelist* für Ereignisse, die periodisch auftreten und als harmlos eingestuft werden können. Es ist beispielsweise in einigen Netzwerken üblich, dass bestimmte Systeme aus Gründen der IT-Sicherheit regelmäßig die vorhandenen Hosts und Dienste aktiv scannen, um auf eventuell kritische Veränderungen in der Infrastruktur aufmerksam zu machen. Derartige Scans würden von HoneySens-Sensoren als Ereignis klassifiziert und immer wieder neu gemeldet werden. Das Filter-Modul erlaubt es, genau solche Ereignisse mit einer Reihe von *Filterregeln* zu beschreiben und zukünftig vom System ignorieren zu lassen.

Nach dem Start des Moduls wird dem Benutzer zunächst eine Liste aller bereits eingerichteten Filter präsentiert. Jeder Filter gilt nur für eine spezifische Gruppe, da IP-Adressen in verschiedenen Teilnetzen mehrfach vorkommen können und das Whitelisting einer Adresse eines bestimmten Teilnetzes normalerweise nicht global gewünscht ist. Aus diesem Grund befindet sich über der Filterliste ein Drop-Down-Menü, mit dem die Gruppe ausgewählt werden kann, deren Filter angezeigt werden sollen. Abbildung 7 zeigt exemplarisch einen solchen Eintrag der Tabelle.



7	Connectify (192.168.209.113)	Whitelist	32	 
---	------------------------------	-----------	----	---

Abbildung 7: Ein Filter mit bisher 32 Übereinstimmungen

Die verschiedenen Spalten dieses Eintrags sind wie folgt zu verstehen:

ID : Die interne Identifikationsnummer für diesen Filter.

Name : Ein frei vom Benutzer zu vergebender Bezeichner zur einfachen Beschreibung und Identifikation der Filterregeln.

Typ : Filter sind immer vom Typ *Whitelist*. Dieses Feld ist für spätere Erweiterungen und Weiterentwicklungen reserviert.

Zähler : Die Zahl gibt an, wie viele mit diesen Filterregeln übereinstimmende Vorfälle bereits beim Server erkannt wurden. Der Zähler dient primär dazu, die Wirksamkeit der Filterregeln zu überprüfen.

Aktionen : Die beiden Buttons in der letzten Spalte dienen zum Bearbeiten der Filterregeln des jeweiligen Filters (Stift-Symbol) und zum unwiderruflichen Entfernen eines Filters.

7.1 Verwaltung von Filterregeln

Mit einem Klick auf den Button *Hinzufügen* öffnet sich ein Dialog, mit dem ein neuer Filter für eine gewünschte Gruppe hinzugefügt werden kann. Jeder einzelne Filter besteht aus einer Reihe von Bedingungen, die *alle* erfüllt sein müssen, damit ein Filter schlussendlich zu einem Ereignis passt und vom System ignoriert wird. Im Dialog wird zunächst für den neuen Filter ein Name vergeben, der informell lediglich zur späteren Identifikation in der Filterlliste dient. Ein anderer Filtertyp außer *Whitelist* kann nicht ausgewählt werden. Dieses Feld ist ein Platzhalter für spätere Weiterentwicklung. Da auch jeder Filter einer einzelnen Gruppe angehört, kann diese im Dialog ebenfalls aus einem Dropdown-Menü ausgewählt werden.

Einzelne Filterbedingungen können mit dem Button *Hinzufügen* an die Liste angehängen werden. Jede Zeile der Tabelle repräsentiert eine einzelne Filterbedingung und setzt sich aus einem zu betrachtenden Attribut und einem Wert, auf den dieses überprüft werden soll, zusammen. Als Attribute stehen zum Filtern zur Verfügung:

Klassifikation : Jedes im System registrierte Ereignis wird automatisch einer von vier Kategorien (Unbekannt, Verbindungsversuch, Honeybot oder Scan) zugeordnet. Die Kategorie *ICMP* ist derzeit nicht in Verwendung, da das System keinen ICMP-Datenverkehr aufzeichnet. Wenn eine Filterbedingung für das Attribut „Klassifikation“ hinzugefügt wird, werden nur solche Ereignisse für diesen Filter berücksichtigt, die in die unter *Wert* ausgewählte Klassifikation fallen.

Quelle : Dieses Attribut bezieht sich auf den Sender der registrierten Ereignisse. Hier kann entweder eine fixe IP-Adresse oder ein ganzer Adressbereich angegeben werden, der von diesem Filter berücksichtigt werden soll. Mit Hilfe des Dropdown-Menüs in der Spalte *Typ* kann ausgewählt werden, welche von beiden Optionen gewünscht wird. Im Feld *Wert* ist schließlich die IP-Adresse oder Adressbereich einzutragen, der gefiltert werden soll.

Ziel : Über dieses Attribut kann eine Portnummer angegeben werden, die von diesem Filter erfasst werden soll. Hierbei ist ausschließlich von *Ziel-Ports* (egal ob UDP oder TCP) die Rede, also die Zieladresse von Paketen, die an die Sensoren geschickt werden. Es ist im Moment nicht möglich, nach Ports seitens des Absenders zu filtern. Als *Wert* ist eine gültige Portnummer zwischen 1 und 65535 einzutragen.

Protokoll : Die Sensoren erfassen TCP- und UDP-Datenverkehr. Falls eine Einschränkung auf eines der Protokolle gewünscht ist, kann dieses im Feld *Wert* spezifiziert werden.

Bedingungen können nach belieben hinzugefügt oder mit dem entsprechenden Button auch wieder entfernt werden. Wenn Sie mit der Zusammenstellung der Bedingungen fertig sind, wird der Filter mit einem Klick auf *Speichern* im System registriert und ist aktiv.

8 Sensoren

Dieses Modul beschäftigt sich mit der Verwaltung der Sensorflotte. Es ist zentraler Anlaufpunkt, um einen Überblick über die vorhandenen Geräte und deren aktuellen Status zu erlangen. Auch verwaltungstechnische Aktionen, wie das Ändern der Basiskonfiguration der Geräte, sowie das Hinzufügen oder Entfernen neuer Geräte kann in diesem Bereich der Anwendung bewerkstelligt werden.

ID ▲	Name	Standort	Firmware	IP-Adresse	Status	Aktionen
4	dud01-bbg	INF 3065	0.2.0	141.76.46.233	✓ Online (vor 6 Minuten)	  

Abbildung 8: Die Sensorliste

In Abbildung 8 ist beispielhaft ein Ausschnitt der Sensorliste dargestellt. Diese kann analog zur Filter- oder auch der Ereignisliste nach Gruppenzugehörigkeit gefiltert werden. Mit Klicks auf die Kopfspalten der Tabelle ist zudem ein Umschalten zwischen verschiedenen Sortiermodi möglich. Die einzelnen Spalten der Tabelle sind wie folgt zu interpretieren:

ID : Die interne Identifikationsnummer für diesen Sensor.

Name : Der bei der Einrichtung eines Sensors vergebene Name. Dieser dient ausschließlich zur Identifikation eines Sensors und unterliegt keinen funktionalen Einschränkungen. Eine spätere Änderung des Namens ist jederzeit möglich.

Standort : Das Standortfeld kann von Benutzern frei genutzt werden, um das Auffinden von Sensoren zu vereinfachen.

Firmware : Jeder Sensor meldet in regelmäßigen Intervallen an den Server seine derzeitige Firmware-Revision. Die letzte auf diesem Weg bekannte Revision wird in diesem Feld angezeigt.

IP-Adresse : Der Sensor informiert den Server mit die regelmäßigen Updates auch über lokale Systeminformationen, darunter die aktuelle IP-Adresse. Die zuletzt bekannte wird in diesem Feld angezeigt.

Status : Dieses Feld gibt den aktuellen Betriebszustand eines Sensors an. Ein grün hinterlegtes Feld mit der Aufschrift *Online* bezeichnet einen Sensor, dessen letzte Statusmeldung beim Server nicht länger in der Vergangenheit liegt als das definierte Rückmeldeintervall. Zusätzlich ist in Klammern angegeben, vor wie vielen Minuten sich das Gerät zuletzt beim Server gemeldet hat. Falls ein Gerät sich innerhalb seines Intervalls nicht gemeldet hat, wird das Statusfeld die Meldung *Timeout* auf rotem Hintergrund anzeigen, sowie den letzten bekannten Meldezeitpunkt anzeigen. Während automatischer Updates der Sensor-Firmware wird zudem zeitweise die Meldung *Update* auf blauem Untergrund angezeigt.

Aktionen : Die drei Aktionsbutton erlauben (1) das Einsehen einer Liste der letzten Statusmeldungen des Sensors, (2) das Editieren der Sensor- und Systemeinstellungen und (3) das Entfernen des gewählten Sensors. **Achtung:** Beim Entfernen eines Sensors werden auch alle von diesem aufgezeichneten Ereignisse gelöscht.

8.1 Hinzufügen und Bearbeiten

Mit einem Klick auf den Button *Hinzufügen* öffnet sich ein Dialogfenster, über das alle Schritte zum Registrieren eines neuen Sensors vorgenommen werden (siehe auch Abbildung 9). Dieser Dialog wird auch beim Bearbeiten eines bereits bestehenden Sensors angezeigt.

Name

Neuer Sensor ✓

Standort

z.B. Raum 312

Gruppe

test

Erreichbarkeit HoneySens-Server

Standard Individuell

Host

honeysens

HTTPS-Port (API)

443

Netzwerkschnittstelle

DHCP Statisch

IP-Adresse und Subnetzmaske werden automatisch vom DHCP-Server bezogen.

MAC-Adresse

Standard Individuell

Es wird die originale MAC-Adresse des verbauten Netzwerkinterfaces genutzt.

HTTP(S)-Proxy

Inaktiv Aktiv

Es kommt kein Proxy-Server zum Einsatz.

Abbildung 9: Dialogoptionen beim Hinzufügen eines Sensors

Jeder Sensor benötigt zunächst einen Namen und einen Standort. Beide können frei gewählt werden und können innerhalb einer HoneySens-Installation auch mehrmals vorkommen. Sie dienen lediglich zur besseren Identifikation und späteren Wiederauffindbarkeit und sollten deshalb einem intern mit anderen Benutzern der gleichen Gruppe vereinbarten Namensschema folgen. Weiterhin gehört jeder Sensor einer Gruppe an, deren Mitglieder das Gerät und die davon aufgezeichneten Ereignisse einsehen können. Hier hat sich eine logische Trennung analog zu physischen Netzen bewährt. Im Dropdown-Feld *Gruppe* ist die Gruppe auszuwählen, der der Sensor angeschlossen werden soll. Diese allgemeinen Angaben werden mindestens benötigt, um einen neuen Sensor registrieren zu können. Die weiteren Dialogbereiche müssen nur bei Bedarf angepasst werden.

Erreichbarkeit HoneySens-Server Die Felder dieses Formulars spezifizieren, wie der Sensor den HoneySens-Server kontaktieren kann. *Host* bezeichnet den über DNS⁶ auflösbaren Hostnamen des Servers. Hierfür ist es zwingend erforderlich, dass der Sensor einen DNS-Server kennt, sprich entweder

⁶Domain Name System

einen solchen per DHCP⁷ vermittelt bekommt oder dieser in einer statischen Netzwerkkonfiguration hinterlegt wird (siehe nachfolgender Abschnitt). Das Feld *HTTPS-Port (API)* gibt außerdem den TCP-Port an, auf dem serverseitig die API über das HTTPS-Protokoll kontaktiert werden kann. Falls der Server nicht bewusst anders konfiguriert wurde ist dies RFC-konform der TCP-Port 443. Für beide Parameter, Host und HTTPS-Port, können globale Standardwerte definiert werden, die für alle neuen Sensoren standardmäßig Anwendung finden sollen. Falls jedoch für einen neuen hinzuzufügenden Sensor abweichende Werte verwendet werden sollen, weil dieser beispielsweise in seinem Netz keinen Zugriff auf einen DNS-Server hat, kann mit einem Klick auf den grauen Button *Individuell* das Formular für eigene Werte freigegeben werden. Es ist an dieser Stelle möglich, die IP-Adresse des HoneySens-Servers (**nicht** aber ein alternativer Hostname) und einen anderen Port zu spezifizieren.

Netzwerkschnittstelle Mit diesen Einstellungen kann das Netzwerkinterface des Sensors konfiguriert werden. Hier besteht zunächst die Wahl zwischen einer automatischen Konfiguration via DHCP (Standard) oder der Vergabe von statischen Netzwerkeinstellungen. Für die DHCP-Option ist selbstverständlich ein entsprechend konfigurierter DHCP-Server im Netz des Sensors nötig, der außerdem auch DNS-Server ausliefert, damit der Sensor den HoneySens-Server kontaktieren kann (Ausnahme: im Abschnitt „Erreichbarkeit“ wurde eine individuelle IP spezifiziert). Falls eine statische Konfiguration des Sensors gewünscht ist, kann dies mit einem Klick auf den Button *Statisch* angezeigt werden. Es werden daraufhin die Felder *IP-Adresse*, *Subnetzmaske*, *Gateway* und *DNS-Server* eingeblendet. Je nach Aufstellung und Erreichbarkeit des HoneySens-Servers müssen die Felder für Gateway und DNS-Server nicht zwingend ausgefüllt werden. Dies hängt von der individuellen Netzarchitektur ab.

Es ist weiterhin möglich, eine individuelle MAC-Adresse zu verwenden. Standardmäßig wird die fixe Hardwareadresse des Sensors genutzt. Falls dies nicht gewünscht ist, beispielsweise wenn die Hardwareadresse noch nicht bekannt ist, aber eine MAC-Adresse zur Konfiguration anderer Netzkomponenten benötigt wird, kann hier eine individuelle MAC-Adresse frei vergeben werden.

HTTP(S)-Proxy Falls der Sensor den HoneySens-Server nicht direkt, sondern ausschließlich über einen (oder eine Kette von) HTTP(S)-Proxy-Server(n) erreichen kann, kann dieser (bzw. das erste Kettenglied) hier eingetragen werden. Nach einem Klick auf den Button *Aktiv* genügt es, die den DNS-Namen oder die IP-Adresse des Proxy-Servers, sowie den verwendeten TCP-Port einzutragen. *Benutzer* und *Passwort* sind optional und werden nur benötigt, wenn der Proxy-Server eine Authentifizierung erfordert. Die Sensoren unterstützen verschiedene Authentifizierungsmethoden, darunter *Basic*, *Digest* und *NTLM*. Die Auswahl der geeigneten Methode erfolgt mit Hilfe eines automatischen Erkennungsverfahrens seitens der Sensorsoftware.

Sensordaten speichern und weitere Schritte Mit einem Klick auf *Speichern* werden alle zuvor gemachten Angaben bestätigt und auf der neue Sensor vom Server gespeichert. Falls ungültige Angaben gemacht wurden, werden diese von der automatischen Formularvalidierung hervorgehoben und müssen korrigiert werden, bevor gespeichert werden kann. Nachdem der Speichervorgang erfolgreich abgeschlossen wurde, werden neben einer entsprechenden Erfolgsmeldung zwei Buttons angezeigt: Einer zum Download der Sensor-Firmware (*Firmware-Download*) und ein weiterer, um die individuelle Sen-

⁷Dynamic Host Configuration Protocol

sorkonfiguration herunterzuladen (*Sensor-Konfiguration*). Zum Verständnis des allgemeinen Konzeptes hinter dieser Trennung empfiehlt sich die Lektüre des zweiten Kapitels *Sensoren* im Abschnitt *Funktionsprinzip*.

Nach dem Download beider Dateien sind folgende nächste Schritte zur Installation des neuen Sensors zu tätigen:

- 1. Installer vorbereiten:** Die heruntergeladene Firmware ist ein Archiv mit der Dateiendung `.tar.gz`. Es beinhaltet zwei Dateien, `firmware.img` und `metadata.xml`. Erstere stellt die eigentliche Sensor-Software dar, während die XML-Datei lediglich zusätzliche Metadaten über die Software, wie beispielsweise die Versionsnummer und eine Kurzbeschreibung enthält. Um den Installer für die Sensor-Firmware zu erstellen, wird zunächst das Archiv entpackt und anschließend die Datei `firmware.img` auf ein Installationsmedium geschrieben. Wenn als Sensoren Geräte vom Typ *BeagleBone Black* oder *BeagleBone Green* zum Einsatz kommen, sind dies microSD-Karten mit einer Größe von *mindestens 4GB*. Zum Schreiben der Firmware-Datei muss zwingend eine Software verwendet werden, die Daten blockorientiert kopiert. Unter Linux-Systemen oder MacOS eignet sich dafür das Programm `dd`, das beispielsweise wie folgt genutzt werden kann:

```
$ dd if=firmware.img of=/dev/mmcblk0 bs=4M
```

Die Parameter `if` (input file) und `of` müssen je nach System angepasst werden, um den richtigen Pfad zum Firmware-Abbild und zum Installationsmedium zu beinhalten. Unter Windows-Systemen empfiehlt sich der Einsatz des kostenlosen Programms *Win32 Disk Imager*⁸, das allerdings nur mit Administratorrechten genutzt werden kann.
- 2. Konfiguration kopieren:** Auf dem mit einem Firmware-Abbild fertig beschriebenen Installationsmedium befinden sich zwei Partitionen: eine große *ext4*-Partiton mit dem Basissystem des Installers und eine 100MB große *FAT*-Partition. Letztere wird beispielsweise unter Windows nach dem Vorbereiten des Installers automatisch als neues Laufwerk angezeigt und kann unter alternativen Betriebssystemen leicht gemountet und beschrieben werden. Da das zuvor geschriebene Firmware-Abbild für alle Installationsmedien identisch ist, um mit einem Schlag möglichst viele Sensoren verteilen zu können, muss im Anschluss noch eine Anpassung dieser Software des späteren Sensors vorgenommen werden. Die beim Hinzufügen des Sensors vorgenommenen Einstellungen zur Netzwerkschnittstelle, Proxy-Servern oder zur Erreichbarkeit des Servers sind alle Bestandteil der separat heruntergeladenen Sensor-Konfiguration. Es gilt deshalb nun noch, diese zweite Datei, bei der es sich ebenfalls um ein `.tar.gz`-Archiv handelt, auf besagte *FAT*-Partition zu kopieren. Anschließend kann das Installationsmedium ordnungsgemäß ausgehängen werden. **Achtung:** Es ist wichtig, dass die Datei *unentpackt* auf das Installationsmedium kopiert wird. Ein vorheriges Entpacken und Kopieren der individuellen Dateien aus dem Archiv wird zu einem nicht funktionsfähigen Sensor führen.
- 3. Sensor installieren:** Das fertig vorbereitete und für den Sensor angepasste Installationsmedium muss nun lediglich noch in das Gerät eingefügt und der Sensor am Zielort mit dem Netzwerk und Strom verbunden werden. Zunächst wird ein Installationsprozess gestartet, in dem der interne Speicher des Sensors geflasht wird. Dieser Vorgang wird bei Geräten der *BeagleBone*-Familie mit einem Laufflicht angezeigt und kann je nach Güte der microSD-Karte bis zu 20 Minuten dauern. Im Anschluss startet das Gerät neu vom frisch installierten internen Speicher, initialisiert sich gemäß der individuellen Sensorkonfiguration und versucht, den HoneySens-Server zu erreichen. Wenn bei diesen Schritten keine Fehler auftraten, startet der Sensor im Anschluss seine HoneyPot-Dienste und wird auch im Frontend des Servers als *Online* angezeigt. Falls es zu Problemen kommt, liegt

⁸<https://sourceforge.net/projects/win32diskimager/>

dies erfahrungsgemäß oft an einer falsch eingetragenen Sensorkonfiguration (beispielsweise ausgelöst durch Tippfehler). Es empfiehlt sich dann, diese zu korrigieren, mit einem Klick auf *Speichern* eine neue Sensorkonfiguration generieren zu lassen und die Schritte 1-3 zu wiederholen.

8.2 Konfiguration

Mit einem Klick auf *Konfiguration* in der Sidebar wird das Modul zur Konfiguration der Sensordienste gestartet. Mit diesem ist es möglich, die Beziehung zwischen Server und Sensoren sowie die angebotenen HoneyPot-Dienste zur Detektion von Einbruchsversuchen zu konfigurieren. Die graphische Oberfläche zur Konfiguration ist zweigeteilt: Im oberen Bereich befinden sich unter der Überschrift *Standard-Sensorkonfiguration* global gültige Standardwerte, die für alle im System hinterlegten Sensoren (aller Gruppen) gelten. Eine Änderung dieser Standardwerte kann lediglich von Administratoren vorgenommen werden. Weniger privilegierte Benutzer können diese Standardwerte hingegen nur einsehen. Zusätzlich ist es jedoch auch möglich, im unteren Bereich Bereich der Seite (Abschnitt *Sensorspezifische Konfiguration*) für einzelne Sensoren eine gesonderte Konfiguration zu hinterlegen. Dies kann genutzt werden, um bestimmte HoneyPot-Dienste nur auf ausgewählten Geräten zu (de-)aktivieren oder neue Firmware-Revisionen auf einzelnen Geräten zu testen, bevor ein globales Update aller Sensoren erfolgt.

Um die globale Konfiguration als Administrator zu modifizieren, genügt es, auf den betreffenden Bereich, beispielsweise *Updateintervall*, zu klicken und im erscheinenden Formular die gewünschten Einstellungen vorzunehmen. Für einzelne Sensoren kann eine neue Konfiguration mit einem Klick auf den Button *Hinzufügen* registriert werden. Es erscheint ein Dialog, der zusätzlich zu den aus der globalen Konfiguration bekannten Optionen noch die Auswahl des betreffenden Sensors und einer gewünschten Firmware-Revision ermöglicht. Wenn die neue Konfiguration wie gewünscht angepasst wurde, kann sie mit einem Klick auf *Speichern* übernommen werden. Der betreffende Sensor wird ab seinem nächsten Polling-Vorgang diese neuen Konfigurationsdaten nutzen. Bereits hinterlegte spezifische Konfigurationen sind in einer Tabelle unter *Sensorspezifische Konfiguration* aufgelistet und können mit Hilfe der entsprechenden Aktionsbuttons modifiziert oder wieder entfernt werden.

Es folgt eine Aufstellung aller Konfigurationsmöglichkeiten:

Updateintervall In diesem Abschnitt kann definiert werden, in welchen zeitlichem Abstand die Sensoren den HoneySens-Server kontaktieren sollen, um ihre Konfigurationsdaten abzufragen und kurze Statusberichte abzuliefern. Das Intervall wird in Minuten spezifiziert und muss im Bereich zwischen 1 und 200 liegen.

Sensor-Dienste Die hier aufgelisteten Dienste bieten die nach außen für Angreifer sichtbare HoneyPot-Funktionalität an und stellen somit das Herzstück der Sensoren dar. Die aufgelisteten Dienste können in beliebigen Kombinationen betrieben werden und sollten dabei auf die jeweiligen Netze, in denen Sensoren zum Einsatz kommen, angepasst werden. Zum Aktivieren oder Deaktivieren einzelner Dienste genügt es, auf die Buttons mit der Aufschrift *AKTIVIERT*, bzw. *DEAKTIVIERT* zu klicken. Aktive Dienste sind zusätzlich farblich grün hinterlegt. Folgende HoneyPot-Dienste stehen zur Auswahl:

Dienst	Beschreibung
recon	<p>Dieser ist ein generischer Dienst, der all die zum Sensor gesendeten Netzwerkpakete aufzeichnet, die von keinem anderen, spezifischeren HoneyPot-Dienst bereits registriert wurden. Recon beschränkt sich dabei allerdings auf IP-Unicast-Pakete (TCP und UDP). Nur Datenverkehr, der direkt an die IP-Adresse des Sensors gesendet wird, kann hiermit aufgezeichnet werden. Um mehr Informationen über potentielle Angriffe gewinnen zu können, zeichnet der Dienst den Datenteil ankommender Pakete mit auf und macht diesen über das Webinterface in der Ereignisanzeige sichtbar (6). TCP-Verbindungen werden zu diesem Zweck vollständig aufgebaut (inklusive dem vom Protokoll verlangten 3-Way-Handshake), das erste Datenpaket vom möglichen Angreifer aufgezeichnet und anschließend die Verbindung wieder ordnungsgemäß geschlossen. Broadcasts, Multicasts und andere Layer-3-Protokolle wie <i>ICMP</i>^a werden hingegen bewusst ignoriert, da mit diesen keine relevanten Angriffsvektoren verbunden sind. Um DoS-Angriffe zu vermeiden, besitzt der recon-Dienst außerdem Erkennungsroutinen für Scanversuche, die einerseits Ereignisse gezielt als <i>Scan</i> an den Server melden und andererseits weitere Pakete vom Quellhost ab diesem Zeitpunkt für eine gewisse Zeitspanne verwerfen.</p> <hr/> <p>^aInternet Control Message Protocol, genutzt für Informations- und Fehlermeldungen</p>
cowrie	<p>Hierbei handelt es sich um ein bestehendes OpenSource-Projekt^a, das einen „Low-Interactive“ HoneyPot für das SSH-Protokoll anbietet. Mit Hilfe dieser Software ist es möglich, dass potentiellen Angreifern von den Sensoren ein simulierter SSH-Server angeboten wird. Es ist möglich, diesen mit herkömmlichen SSH-Clients (z.B. <i>OpenSSH</i>) zu kontaktieren, sich mit Benutzerdaten zu authentifizieren und, falls die „korrekten“ Zugangsdaten übergeben wurden, auch eine simulierte Shell-Session zu starten. Eine Übernahme des Sensors durch diese simulierte, letztendlich funktionslose Shell ist allerdings abgesehen von potentiellen Bugs innerhalb von cowrie selbst, nicht möglich. Sehr wohl wird aber jegliche Interaktion des Angreifers mit dem Sensor aufgezeichnet, darunter alle Befehle, die dieser in der Shell absetzt und alle Benutzernamen und Passwörter, mit denen eine Authentifizierung via SSH angestrebt wird. Für weitere Details empfiehlt es sich, die Website des cowrie-Projektes zu besuchen. Im Rahmen des HoneySens-Projektes wurde cowrie um ein Modul erweitert, das aufgezeichnete Daten in einem einheitlichen Format an den Server weiterleitet. Alle gesammelten Daten können in den Ereignisdetails eines jeden Vorfalls eingesehen werden (siehe auch 6).</p> <hr/> <p>^ahttps://github.com/micheloosterhof/cowrie</p>

dionaea	<p>Als ebenfalls frei verfügbares OpenSource-Projekt^a und auch Low-Interaction-Honeypot kann Dionaea eine ganze Reihe von Diensten mit unterschiedlichem Interaktionsgrad für Angreifer simulieren. Das SMB/CIFS-Protokoll, mit dem in Windows-Domänen Datei- und Druckerfreigaben genutzt werden können, ist seither aufgrund diverser Sicherheitsprobleme ein beliebtes Ziel für Schadsoftware und auch menschliche Angreifer, um Schäden an IT-Systemen herbeizuführen. Dionaea ist auf die Abbildung dieser Protokollfamilie mit zugehörigen Sicherheitslücken spezialisiert und ist deshalb insbesondere beim Einsatz innerhalb von Windows-Domänen sinnvoll, um beispielsweise auf Netzwerkwürmer aufmerksam zu werden, die sich mit Hilfe solcher Exploits zu verbreiten versuchen. Alle weiteren Protokolle, die von Dionaea upstream unterstützt werden, sind in der HoneySens-Firmware deaktiviert, da diese oft nur minimale Interaktivität erlauben und somit kaum zusätzlichen Informationsgewinn im Vergleich zum zuvor beschriebenen recon-Dienst mit sich bringen. Auch Dionaea wurde analog zu cowrie um ein Reporting-Modul erweitert, das aufgezeichnete Daten an den HoneySens-Server weiterleitet, so dass diese direkt im Webinterface ausgewertet werden können.</p> <hr/> <p>^ahttps://github.com/rep/dionaea</p>
---------	---

Firmware Die Auswahl einer gewünschten Firmware-Revision ist ausschließlich im Kontext einer individuellen Sensorkonfiguration sichtbar. Somit ist es möglich, einzelne Sensoren gezielt mit einer vom globalen Standard abweichenden Firmware zu versehen. Dies kann beispielsweise hilfreich sein, um eine neue Version erst ausführlich mit wenigen Sensoren testen zu können, bevor ein globales Update freigegeben wird. Die systemweit gültige Firmware wird im gleichnamigen Modul ausgewählt (siehe Abschnitt 8.3).

8.3 Firmware

Als *Firmware* wird die Software bezeichnet, mit der Sensoren betrieben werden. Derzeit werden ausschließlich Geräte vom Typ *BeagleBone Black* und *BeagleBone Green* als Hardwareplattform für Sensoren unterstützt. Beide sind miteinander kompatibel, so dass die gleiche Firmware für beide Sensortypen genutzt werden kann.

Damit eine bestimmte Firmware für Sensoren genutzt werden kann, muss diese zunächst beim Server registriert werden. Hierzu genügt es, ein Firmware-Archiv im korrekten Format (tar.gz-Archiv mit Firmware- und METadaten) mit Hilfe des Webinterfaces auf den Server zu laden. Das Firmware-Modul in der Verwaltungsoberfläche präsentiert nach dem Aufruf eine Liste aller auf dem Server registrierten Firmware-Revisionen und einen Button mit der Aufschrift *Hinzufügen*, mit dem ein neues Firmware-Archiv im Dateisystem für den Upload ausgewählt werden kann. Nach dem Upload eines dieser mehreren hundert Megabyte großen Archive finden eine Reihe von serverseitigen Tests statt, mit denen überprüft wird, ob es sich tatsächlich um eine gültige Firmware handelt. Wenn diese Tests erfolgreich waren, wird der Liste der registrierten Revisionen ein weiterer Eintrag mit einer Kurzbeschreibung hinzugefügt. Eine Zeile der Tabelle ist typischerweise gelb hinterlegt und bietet als „Aktionen“ ausschließlich den Download der betreffenden Firmware an. Bei diesem Eintrag handelt es sich um die global gültige Stan-

dardfirmware, die von allen Sensoren als aktuell betrachtet wird (außer es wurden für einzelne Sensoren gezielt alternative Firmwarekonfigurationen hinterlegt, siehe hierzu 8.2).

Mit den Buttons auf der rechten Seite kann eine gewählte Firmware erneut heruntergeladen werden, um beispielsweise Installationsmedien schon im Vorfeld mit dem generischen Installer vorzubereiten und zu einem späteren Zeitpunkt die Sensorkonfiguration hinzuzufügen. Weiterhin kann mit dem Button, der einen nach oben deutenden Pfeil darstellt, eine bestimmte Firmware zum globalen Standard erhoben werden. Wenn diese geändert wird, werden im Anschluss alle Sensoren ein vollautomatisches Update auf die betreffende Version durchführen (ausgenommen Sensoren mit individueller Firmware-Konfiguration, siehe 8.2). Zuletzt ist es auch noch möglich, Firmware-Revisionen wieder rückstandslos vom Server zu entfernen. Diese Operation kann jedoch nur für Firmware durchgeführt werden, die derzeit nicht für einen der Sensoren vorgesehen ist.

9 Benutzer- und Gruppenverwaltung

Das Modul *Benutzer und Gruppen* fügt dem HoneySens-System Funktionen zur Mandantenverwaltung hinzu. Wie eingangs bereits erklärt (siehe III), gibt HoneySens verschiedenen Nutzergruppen Raum für eigene, selbst verwaltbare Sensoren, Filterregeln, Konfigurationen usw. Je nach Benutzerrolle können diese Ressourcen von Gruppenmitgliedern eingesehen oder auch modifiziert werden. Das hier besprochene Modul erlaubt es, Benutzer und Gruppen zu verwalten und ist ausschließlich Nutzern mit Administratorstatus zugänglich.

Die Benutzer- und Gruppenverwaltung präsentiert sich zweiseitig: in der linken Spalte sind alle im System registrierten Benutzer aufgelistet, während die rechte Spalte alle Gruppen darstellt.

9.1 Benutzer

Die Spalten der Benutzertabelle sind selbsterklärend: Jeder Nutzer wird mit seiner internen Identifikationsnummer, dem Login-Namen, seiner Rolle (Administrator, Manager oder Beobachter) und Buttons zum Bearbeiten oder Entfernen eines Datensatzes gelistet. Ein neuer Nutzer kann mit dem gleichnamigen Button hinzugefügt werden.

Das Formular zum Bearbeiten oder Hinzufügen von Benutzern ist in Abbildung 10 abgebildet. Jedem Benutzer muss einen im System nur einmal vorkommenden Name vergeben werden, der gleichzeitig auch für das Login zum Einsatz kommt. Weiterhin benötigt jeder Nutzer ein Passwort, das mindestens eine Länge von sechs Zeichen umfasst. Zusätzlich ist es ratsam, eine

Benutzer hinzufügen Speichern Abbrechen

Name
Benutzername

Passwort
Passwort

E-Mail
E-Mail-Adresse

Rolle
Beobachter

Abbildung 10: Dialog zum Hinzufügen von Nutzern

gültige E-Mail-Adresse zu vergeben, denn der HoneySens-Server kann auf Wunsch auch Benachrichtigungen per Mail versenden. Von großer Bedeutung ist außerdem noch die Auswahl der korrekten Rollen für den Benutzer. Eine Übersicht über die Unterschiede zwischen den verfügbaren Rollen wird im Abschnitt III dieses Handbuchs gegeben. **Achtung:** Ein Benutzer kann derzeit keine gruppenspezifischen Rollen annehmen, d.h. ein Nutzer mit der Rolle *Manager* übt für **alle** Gruppen, denen er zugewiesen ist, diese Rolle aus.

Um sicherzustellen, dass das System administrierbar bleibt, muss es immer mindestens einen Administrator geben. Hierfür ist der Nutzer mit der ID 1 reserviert. Dieser ist in der Benutzerliste gesondert markiert und kann weder entfernt noch dessen Rolle geändert werden.

9.2 Gruppen

Jeder Gruppe sind eine Reihe von Ressourcen, darunter Benutzer, Sensoren und Ereignisfilter, zugeordnet. In der tabellarischen Gruppenübersicht der Benutzer- und Gruppenverwaltung sind alle im System registrierten Gruppen mit der zugehörigen Anzahl an Benutzern und Sensoren aufgelistet. Mit Hilfe des Buttons *Hinzufügen* können neue Gruppen eingetragen werden. Zusätzlich erlaubt die Spalte *Aktionen*, einzelne Gruppen zu bearbeiten oder zu entfernen. Sobald eine neue Gruppe im System registriert wurde, kann diese in allen anderen Bereichen der Anwendung, beispielsweise dem Sensor-Modul zum Filtern und Hinzufügen von Sensoren, genutzt werden.

Der Dialog zum Hinzufügen und Bearbeiten von Gruppen macht zunächst das Eintragen eines Namens für die Gruppe erforderlich. Dieser ist abgesehen von einer Beschränkung auf bestimmte Sonderzeichen keinen zusätzlichen Bestimmungen unterworfen. Der restliche Dialog unterteilt sich zum Einen in eine Liste aller Benutzer, die Teil dieser Gruppe sind und eine Liste von Kontakten, die E-Mail-Benachrichtigungen für Ereignisse an den Sensoren der Gruppe erhalten sollen. Zum Hinzufügen eines Benutzers zur Gruppe ist es erforderlich, dass dieser zuvor separat angelegt wurde (siehe 9.1). Nach einem Klick auf den Button *Hinzufügen* wird ein Dropdown-Menü sichtbar, in dem alle Benutzer aufgelistet sind, die noch nicht Mitglieder der betreffenden Gruppe sind. Es genügt, den Benutzernamen auszuwählen, der zur Gruppe hinzugefügt werden soll. Es entsteht anschließend ein neuer Eintrag in der gruppenspezifischen Benutzertabelle, über den Nutzer auch mit dem entsprechenden Button wieder aus der Gruppe entfernt werden können.

Es können beliebig viele gruppenspezifische Kontakte für automatisierte E-Mail-Benachrichtigungen spezifiziert werden. Hier hängt ein Klick auf den Button *Hinzufügen* einen neuen Kontakt-Eintrag an. Für diesen muss zunächst ausgewählt werden, wer per E-Mail zu informieren ist. In der Spalte *Typ* kann hierfür zwischen den Optionen *E-Mail* und *Benutzer* ausgewählt werden. Erstere erlaubt die Angabe einer beliebigen E-Mail-Adresse, an die Benachrichtigungen zugestellt werden sollen. Letztere nutzt hingegen die für die einzelnen Benutzer beim Anlegen hinterlegten E-Mail-Adressen, weshalb hier das Auswählen eines Nutzernamens, der Benachrichtigungen erhalten soll, genügt. Die Spalte *Kontakt* dient dann zur Spezifikation des konkreten Adressaten (separate E-Mail oder Benutzer). Nachdem der Adressat bestimmt wurde, muss weiterhin definiert werden, was für Notifikation dieser erhalten soll. Dies wird über die Spalte *Nachricht für* abgehandelt. Darin befinden sich zwei Buttons, die jeweils an- oder ausgeschaltet werden können:

Wöchentl. Zusammenfassung : Wenn diese Option aktiv ist, erhält der Adressat einmal pro Woche (fix am Montag morgen) eine kurze statistische Zusammenfassung über alle Ereignisse, die in der vorangegangenen Woche aufgetreten sind.

Kritische Ereignisse : Das Aktivieren dieser Option bewirkt, dass der Adressat bei jedem „kritischen“ Ereignis eine E-Mail mit den Details zum jeweiligen Vorfall. Als kritisch werden Kontaktversuche zu den beiden HoneyPot-Diensten *cowrie* (SSH) und *dionaea* (SMB/CIFS) eingeordnet.

Es ist selbstverständlich auch möglich, beide Optionen gleichzeitig zu aktivieren. **Hinweis:** Damit die E-Mail-Benachrichtigung genutzt werden kann, ist es zusätzlich erforderlich, einen gültigen systemweiten Mail-Server hinterlegt zu haben (siehe auch Kapitel 10).

Mit einem Klick auf *Speichern* werden alle Änderungen an der Gruppenkonfiguration gespeichert.

10 Systemeinstellungen

Das Modul *System* ist ausschließlich für Administratoren zugänglich und erlaubt es, global gültige Standardeinstellungen vorzunehmen, die die Funktionalität des Gesamtsystems betreffen. Dazu gehören zum einen eine Reihe von Wartungsoptionen, die Definition des Server-Endpunkts aus Sensorsicht und die E-Mail-Konfiguration.

Wartung Zur Wartung des Systems existierende folgende Operationen, die jeweils mit einem Klick auf die Aktion und dann auf den Button neben der aufklappenden näheren Beschreibung durchgeführt werden können. **Achtung:** Diese Operationen führen invasive Veränderungen an der Datenbank durch und können unter Umständen die Funktionsfähigkeit des Systems beeinträchtigen. Legen Sie zuvor ein Backup des HoneySens-Datencontainers an.

Datenbank zurücksetzen : Diese Operation entfernt alle Inhalte aus der Datenbank und setzt das System auf den Anfangszustand zurück. Es existiert dann nur noch ein Standardnutzer namens *admin* mit dem Passwort *admin*. Es ist dann zunächst erforderlich, mindestens eine neue Benutzergruppe anzulegen und eine Firmware hochzuladen, um neue Sensoren registrieren zu können. Daten von bestehenden Sensoren gehen verloren - es ist zudem eine Neuinstallation aller Sensoren erforderlich. Erhalten bleiben die Server-Endpunktdaten und die E-Mail-Konfiguration im Modul *System*, da diese außerhalb der Datenbank in einer separaten Konfigurationsdatei abgelegt werden.

Ereignisse entfernen : Mit dieser Operation wird die Ereignisliste vollständig geleert. Alle anderen Ressourcen, auch Ereignisfilter, bleiben allerdings erhalten. Das System bleibt nach dieser Operation funktionstüchtig.

Schema-Update : Diese Aktivität aktualisiert das Datenbankschema, was nach Updates erforderlich sein kann. Da mit der HoneySens-Version 0.2.0 ein Update-Assistent eingeführt wurde, der u.a. auch diese Operation automatisch durchführt, ist ein manuelles Aktivieren der Schema-Aktualisierung normalerweise nicht mehr erforderlich. Um bei eventuell auftretenden Problemen mit dem Assistenten eine Alternative anzubieten, wird diese Operation übergangsweise noch beibehalten, aber in einer zukünftigen Version entfernt.

Server-Endpunkt In diesem Formular wird für alle Sensoren definiert, wie sie den HoneySens-Server erreichen können. Die hier eingetragenen Werte stehen im Dialog zum Hinzufügen und Bearbeiten von Sensoren als Standardwerte zur Verfügung (siehe Kapitel 8.1), können in diesem aber auch für einzelne Sensoren separat angepasst werden.

Sensoren werden versuchen, eine HTTPS-Verbindung auf dem angegebenen TCP-Port zum Host mit dem angegebenen Namen aufzubauen. Dazu ist es zunächst erforderlich, dass die Sensoren den Hostnamen des Servers auch per DNS auflösen können. Zusätzlich ist es zwingend notwendig, dass als Hostname hier der *Common Name* des genutzten TLS-Serverzertifikates eingetragen wird. Falls dieser nicht mit dem tatsächlichen Hostnamen übereinstimmt, was beispielsweise bei Setups mit selbstsigniertem Zertifikat der Fall ist (ein solches wird beim Anlegen einer neuen Serverinstanz als Backup-Verfahren automatisch erzeugt), muss als systemweiter Host der Common Name des Zertifikates eingetragen werden. Zusätzlich ist es dann erforderlich, dass bei der Einrichtung der Sensoren **nicht** dieser globale Wert als Server-Endpunkt genutzt wird (da diese den Namen in der Regel nicht per DNS auflösen können), sondern dass *individuell* die IP-Adresse, unter der der Server erreichbar ist, spezifiziert wird. Dies ist im Fall des selbstsignierten Zertifikates für jeden Sensor gesondert erforderlich.

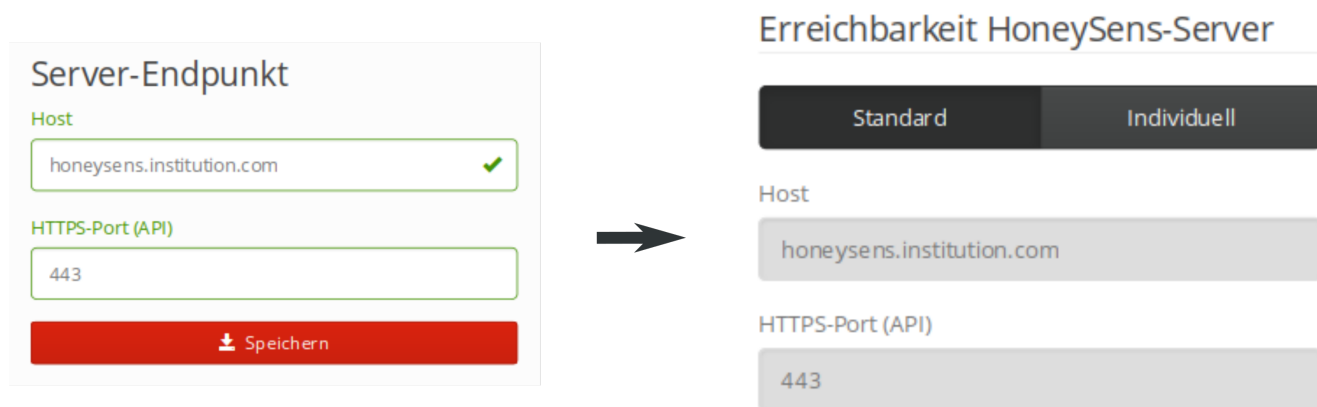


Abbildung 11: Definition des Server-Endpunktes unter Nutzung eines gültigen Zertifikats

Abbildung 11 veranschaulicht diesen Umstand: Wenn der Common Name des für den Server genutzten TLS-Zertifikats von den Sensoren per DNS auflösbar ist, können beim Hinzufügen von Sensoren die systemweiten Standardwerte übernommen werden. Im Beispiel besitzt der Server ein gültiges TLS-Zertifikat für die Domain `honeysens.institution.com`, die auch von den Sensoren entsprechend via DNS aufgelöst werden kann. Auch wenn dieser systemweite Wert jemals geändert werden sollte, aktualisieren die Sensoren diesen Wert automatisch. Dieses Setup stellt zugleich den vorgesehenen Einsatzzweck von HoneySens dar. Falls hingegen keine gesonderten Zertifikate für die Domain des HoneySens-Servers bei der Installation zur Verfügung gestellt werden, kommt automatisch ein selbstsigniertes Zertifikat zum Einsatz, das für jede Installation neu erzeugt wird. In diesem Fall ist es erforderlich, als globalen Endpunkten den im automatisch erzeugten Zertifikat hinterlegten Common Name als Endpunkt einzutragen, damit die Sensoren mit dem Server kommunizieren können. Da dieser vom System zufällig vergebene Common Name kein von den Sensoren auflösbarer Domainname ist, muss zufällig für jeden Sensor noch separat die IP-Adresse, unter der er den HoneySens-Server erreichen kann, spezifiziert werden. Diesen Umstand veranschaulicht beispielhaft Abbildung 12.

Hier ist `a7f8dds24g14d` der vom Server automatisch erzeugte Common Name des Zertifikates (typischerweise die ID der Docker-Instanz) und `10.0.1.6` die IP-Adresse, aus der ein Sensor den Server

The image shows two screenshots of the HoneySens configuration interface. The left screenshot, titled "Server-Endpoint", shows a form with two input fields: "Host" containing "a7f8dds24g14d" and "HTTPS-Port (API)" containing "443". A red "Speichern" button is at the bottom. The right screenshot, titled "Erreichbarkeit HoneySens-Server", shows a "Standard" button selected, with the "Host" field containing "10.0.1.6" and the "HTTPS-Port (API)" field containing "443". Both fields have green checkmarks. An arrow points from the left screenshot to the right one.

Abbildung 12: Definition des Server-Endpunktes mit selbstsigniertem Zertifikat

erreichen kann.

SMTP-Konfiguration HoneySens kann automatisiert E-Mails versenden: Entweder, um Nutzer über kritische Vorfälle zu informieren oder zur Bereitstellung von wöchentlichen Zusammenfassungen über alle binnen der letzten sieben Tage aufgetretenen Ereignisse. Die Konfiguration, wann und an wen welche E-Mails verschickt werden sollen, ist Teil der Benutzerverwaltung und wird in Kapitel 9.2 beschrieben. Als Voraussetzung, damit dieser Mechanismus genutzt werden kann, ist allerdings zunächst noch die Konfiguration eines SMTP-Server notwendig, über den Mails an ihre Adressaten verschickt werden können. Das vorliegende Formular bietet hierfür die folgenden Felder an:

Server : Der via DNS auflösbare Hostname oder die IP-Adresse des zu nutzenden E-Mail-Servers.

Absender : Die E-Mail-Adresse, die als Absender bei vom System generierten Nachrichten eingetragen werden soll.

Benutzer und Passwort : *optional*, wenn der Server Authentifizierung erfordert.

Teil IV

HowTos

Dieser Abschnitt stellt exemplarisch vor, wie einige häufig durchzuführende Arbeitsschritte mit dem System durchgeführt werden können. Jedes Kapitel stellt einen solchen Workflow vor, indem zunächst das zu erreichende Ziel, die dafür notwendigen Voraussetzungen und anschließend die einzeln durchzuführenden Schritte erklärt werden.

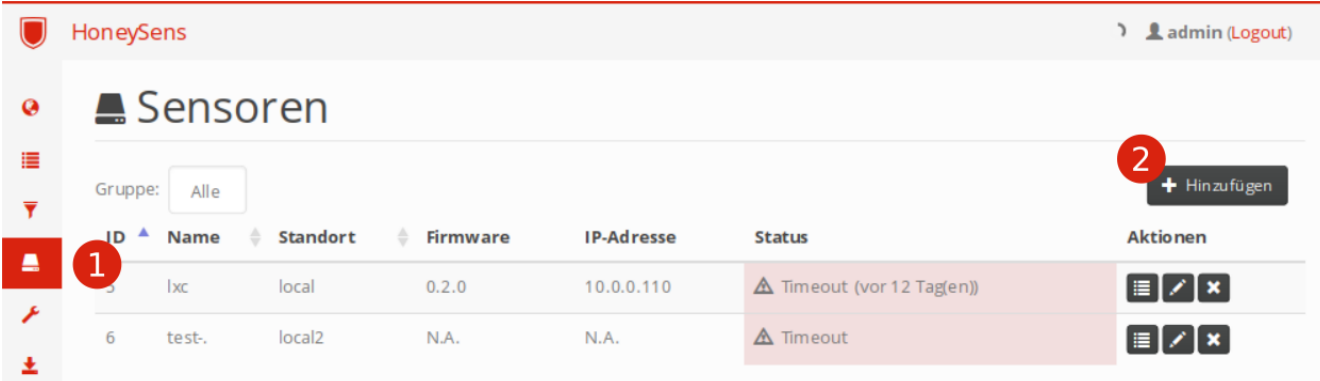
11 Einrichten eines neuen Sensors

Ziel Es soll ein neuer Sensor vom Typ *BeagleBone Black* zu einer Gruppe hinzugefügt und ans Netz gebracht werden.

Voraussetzungen

- Nutzer ist mit einem Manager- oder Administratoraccount angemeldet (Kapitel 3)
- Nutzer gehört der Gruppe an, in die der neue Sensor eingefügt werden soll (Kapitel 9.2)
- Gültige Firmware ist auf dem Server als Standardfirmware registriert (Kapitel 8.3)
- Der globale Server-Endpoint ist korrekt definiert (Kapitel 10)
- **Hardware:** BeagleBone Black, microSD-Karte (mind. 4GB), Kabel zur Netzwerk- und Stromversorgung

Vorgehensweise Zunächst muss der neue Sensor auf dem Server mit Hilfe der Webanwendung registriert werden. Hierfür wird zuerst das Modul *Sensoren* in der Sidebar ausgewählt (1) und über der erscheinenden Liste der Button *Hinzufügen* angeklickt (2).



The screenshot shows the HoneySens web interface. The top navigation bar includes the HoneySens logo and a user profile for 'admin (Logout)'. The main content area is titled 'Sensoren'. On the left sidebar, the 'Sensoren' icon is highlighted with a red circle '1'. In the main content area, there is a 'Gruppe:' dropdown menu set to 'Alle' and a '+ Hinzufügen' button highlighted with a red circle '2'. Below this is a table of sensors:

ID	Name	Standort	Firmware	IP-Adresse	Status	Aktionen
5	lxc	local	0.2.0	10.0.0.110	⚠ Timeout (vor 12 Tag(en))	[List] [Edit] [Delete]
6	test-	local2	N.A.	N.A.	⚠ Timeout	[List] [Edit] [Delete]

Es erscheint ein Formular, in dem alle für den neuen Sensor relevanten Einstellungen vorgenommen werden. Zunächst gilt es, einen Namen für den Sensor zu vergeben (1). Dieser dient lediglich dazu, das Gerät

über die Webanwendung und in der Ereignisliste leichter identifizieren zu können. Ebenso verhält es sich mit der Angabe des Standortes (2). Hier kann ein beliebiger String eingetragen werden, der das Auffinden des Geräts vor Ort erleichtern soll - beispielsweise eine Gebäude- oder Raumnummer. Außerdem muss mit einer Dropdown-Liste noch spezifiziert werden, welcher Gruppe das Gerät angeschlossen werden soll (3).

+ Sensor hinzufügen

Name
Sensorname 1

Standort
z.B. Raum 312 2

Gruppe
test 3

Erreichbarkeit HoneySens-Server
Standard 4 Individuell

Host
honeysens

HTTPS-Port (API)
443

Netzwerkschnittstelle
DHCP 5 Statisch

IP-Adresse und Subnetzmaske werden automatisch vom DHCP-Server bezogen.

MAC-Adresse
Standard 6 Individuell

Es wird die originale MAC-Adresse des verbauten Netzwerkinterfaces genutzt.

HTTP(S)-Proxy
Inaktiv 7 Aktiv

Es kommt kein Proxy-Server zum Einsatz.

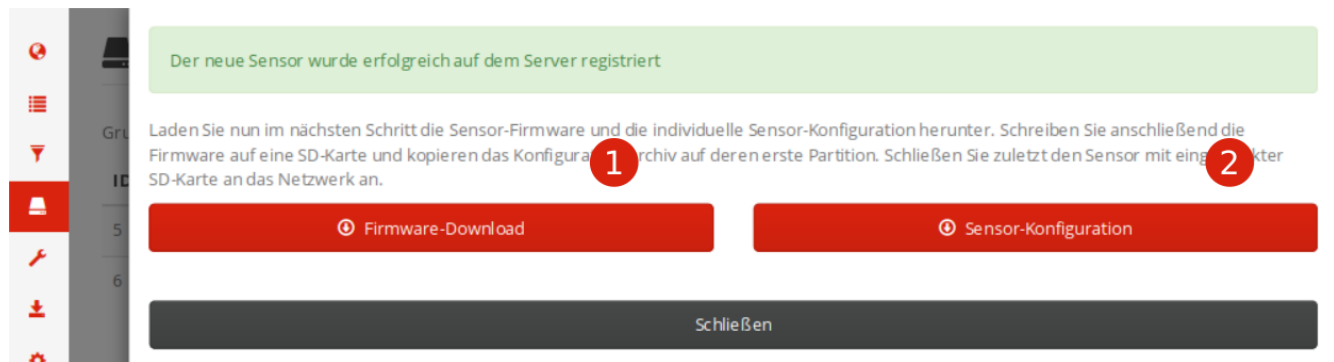
Abbrechen Speichern 8

Nach diesen allgemeinen Angaben können bei Bedarf noch weitere Einstellungen unter den Punkten „Erreichbarkeit HoneySens-Server“, „Netzwerkschnittstelle“ und „HTTP(S)-Proxy“ vorgenommen werden.

Falls der HoneySens-Server mit den unter *Host* und *HTTPS-Port* angegebenen globalen Daten vom neuen Sensor nicht erreichbar sein wird (weil beispielsweise im Zielnetz kein DNS-Server zur Auflösung des Hostnamens zur Verfügung steht), kann mit einem Klick auf *Individuell* alternativ eine IP-Adresse angegeben werden (4). Wenn im Zielnetz kein DHCP-Server zur Verfügung steht, kann unter *Netzwerkschnittstelle* mit einem Klick auf *Statisch* auch eine individuelle IP-Adresse mit zugehöriger Netzmaske, Gateway und DNS-Server spezifiziert werden. Außerdem kann bei Bedarf eine individuelle MAC-Adresse festgelegt werden (6). Standardmäßig kommt die originale Hardware-MAC-Adresse des Sensors unverändert zum Einsatz. Zuletzt ist es außerdem noch möglich, bei Bedarf HTTP(S)-Proxy-Server zu definieren. Dies ist nur nötig, wenn der HoneySens-Server nicht über normale Routen erreichbar ist und der einzige Weg zwingend über Proxy-Server führt. Nach einem Klick auf *Aktiv* erscheint ein Formular, in dem die Adresse des zu verwendenden Proxy-Servers und bei Bedarf auch Nutzernamen und Passwort festgelegt werden können (7). Die Sensorsoftware wird automatisch versuchen, das korrekte Authentifizierungsverfahren zu finden.

Wenn alle Einstellungen korrekt vorgenommen wurden, kann mit einem Klick auf *Speichern* fortgefahren

werden (8).



Im anschließend erscheinenden Dialog werden zwei Dateien zum Download angeboten: Die generische Sensor-Firmware (1) und die an den Sensor angepasste Konfiguration (2).

Das Firmware-Archiv beinhaltet ein Abbild eines Installers, das auf ein Installationsmedium (microSD-Karte) geschrieben werden muss und für alle Sensoren identisch ist. Dieses Archiv kann alternativ auch schon vor dem Einrichten eines Sensors im Modul *Firmware* gesondert heruntergeladen werden. Damit ein Sensor aber nach der Installation mit seinen zuvor eingegebenen Konfigurationsdaten (Netzwerkkonfiguration, Erreichbarkeit Server usw.) versorgt werden kann, ist es vor der Installation noch notwendig, die Sensor-Konfiguration (2) auf das Installationsmedium zu kopieren.

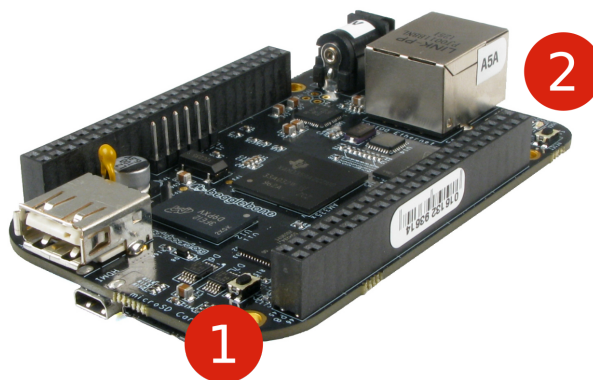
Hinter dem Firmware-Download verbirgt sich ein Archiv mit der Endung `.tar.gz`, das von den meisten Archivierungsprogrammen entpackt werden kann. Es beinhaltet zwei Dateien: den Installer selbst (`firmware.img`) und eine Datei mit beschreibenden zugehörigen Metadaten (`metadata.xml`), die für die Installation ignoriert werden kann (`firmware.img`) und eine Datei mit beschreibenden zugehörigen Metadaten (`metadata.xml`), die bei der Installation ignoriert werden kann. Zunächst gilt es, den Installer auf eine microSD-Karte zu schreiben. Unter unixoiden Betriebssystemen wie *Linux* oder *OSX* kann dies mit Hilfe des Tools `dd` erfolgen. Unter *Windows* kann beispielsweise *WinDiskImager* genutzt werden. Nachfolgend exemplarisch das Auspacken und Schreiben der Firmware auf die microSD-Karte `/dev/mmcblk0` unter *Linux*:

```
$ tar xzf s1.tar.gz
$ ls
. .. firmware.img metadata.xml s1.tar.gz
$ dd if=./firmware.img of=/dev/mmcblk0 bs=4M
```

Sobald das Abbild erfolgreich auf das Installationsmedium geschrieben wurde, befindet sich darauf neben dem eigentlichen Installer auch eine 100 Megabyte große FAT-Partition, die zur Konfiguration genutzt wird unter beispielsweise unter *Windows* automatisch eingehangen wird. Auf diese Partition muss zuletzt noch das Konfigurationsarchiv (2) kopiert werden. **Hinweis:** Das Archiv mit den Konfigurationsdaten darf zuvor **nicht** entpackt werden! Dies kann unter unixoiden Systemen beispielsweise folgendermaßen aussehen:

```
$ mount /dev/mmcblk0p1 /mnt
$ cp s1.tar.gz /mnt
$ umount /dev/mmcblk0p1
```

Die Vorbereitungen für die microSD-Karte als Installationsmedium sind an dieser Stelle abgeschlossen. Zuletzt gilt es, das Installationsmedium in den nebenstehend abgebildeten Sensor einzulegen, der Slot für die microSD-Karte befindet sich an Position (1). Das Gerät wird nach dem Einschalten von diesem Medium booten und die automatische Installation anstoßen, während der die Sensor-Firmware auf den internen Speicher des BeagleBone geschrieben wird. Im Anschluss erfolgt ein automatischer Neustart und das Gerät ist einsatzbereit. Zuvor ist noch zu berücksichtigen, dass die Stromversorgung und Netzwerkverbindung angeschlossen sind. Die entsprechenden Anschlüsse befinden sich an Position (2). Mit dem Einstecken der Stromversorgung wird der Sensor automatisch starten, die Installation durchführen und sich anschließend beim Server registrieren.



Die Installation eines Sensors dauert je nach Geschwindigkeit des verwendeten Installationsmediums etwa 20 bis 30 Minuten. Um dies zu veranschaulichen, zeigen die vier blauen LEDs an Seite (2) während des Sensors während dieses Vorgangs ein Lauflicht an. Abschließend versucht der Sensor, den Server zu kontaktieren. Wenn dieser Kontakt erfolgreich war, wechselt der Status des Sensors in der Sensorübersicht auf *online*. Der Sensor ist anschließend einsatzbereit und meldet verdächtige Ereignisse an den Server.

The screenshot shows the 'Sensoren' (Sensors) page in the HoneySens web interface. The table below shows a sensor with ID 5, name 'lxc', location 'local', and status 'Timeout'.

ID	Name	Standort	Firmware	IP-Adresse	Status	Aktionen
5	lxc	local	N.A.	N.A.	Timeout	[List] [Edit] [Delete]



The screenshot shows the 'Sensoren' (Sensors) page in the HoneySens web interface after the sensor has successfully connected. The table below shows the same sensor with ID 5, name 'lxc', location 'local', firmware '0.2.0', IP address '10.0.0.110', and status 'Online (vor 0 Minuten)'.

ID	Name	Standort	Firmware	IP-Adresse	Status	Aktionen
5	lxc	local	0.2.0	10.0.0.110	Online (vor 0 Minuten)	[List] [Edit] [Delete]

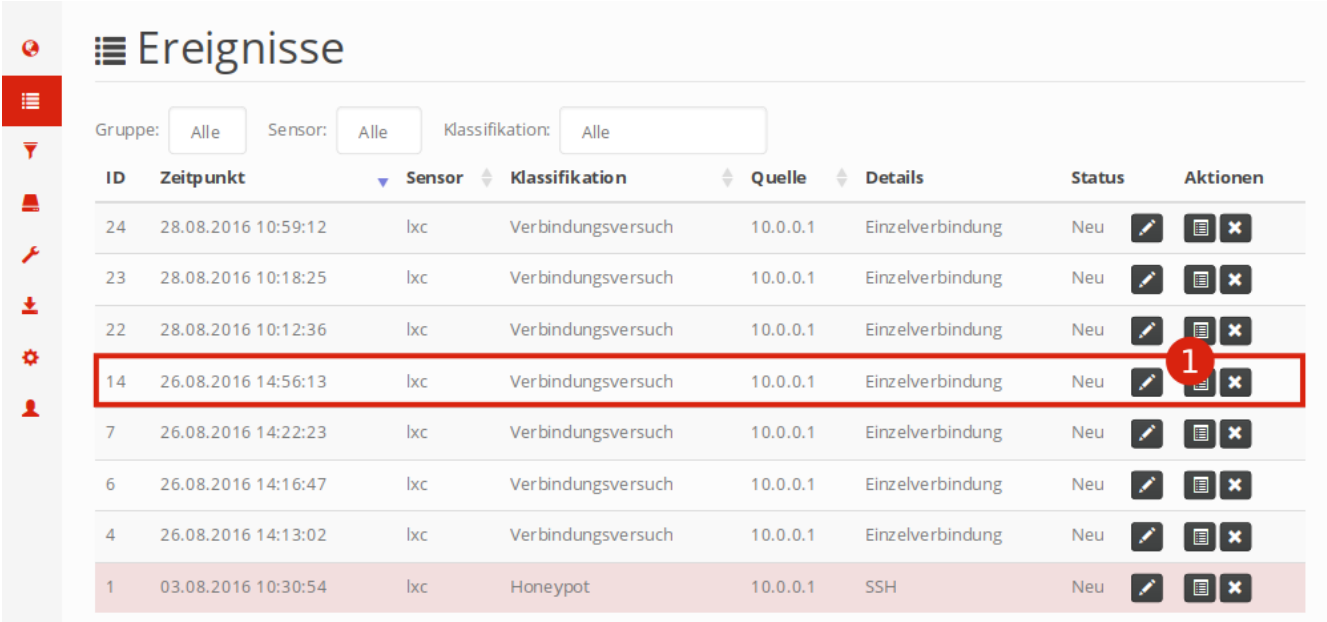
12 Ereignisfilter anlegen

Ziel Für wiederkehrende, als harmlos einzustufende Ereignisse einer bestimmten Quell-IP-Adresse soll ein Filter eingerichtet werden, damit zukünftige Vorfälle dieser Art nicht mehr gesondert aufgezeichnet werden (sog. *Whitelisting*).

Voraussetzungen

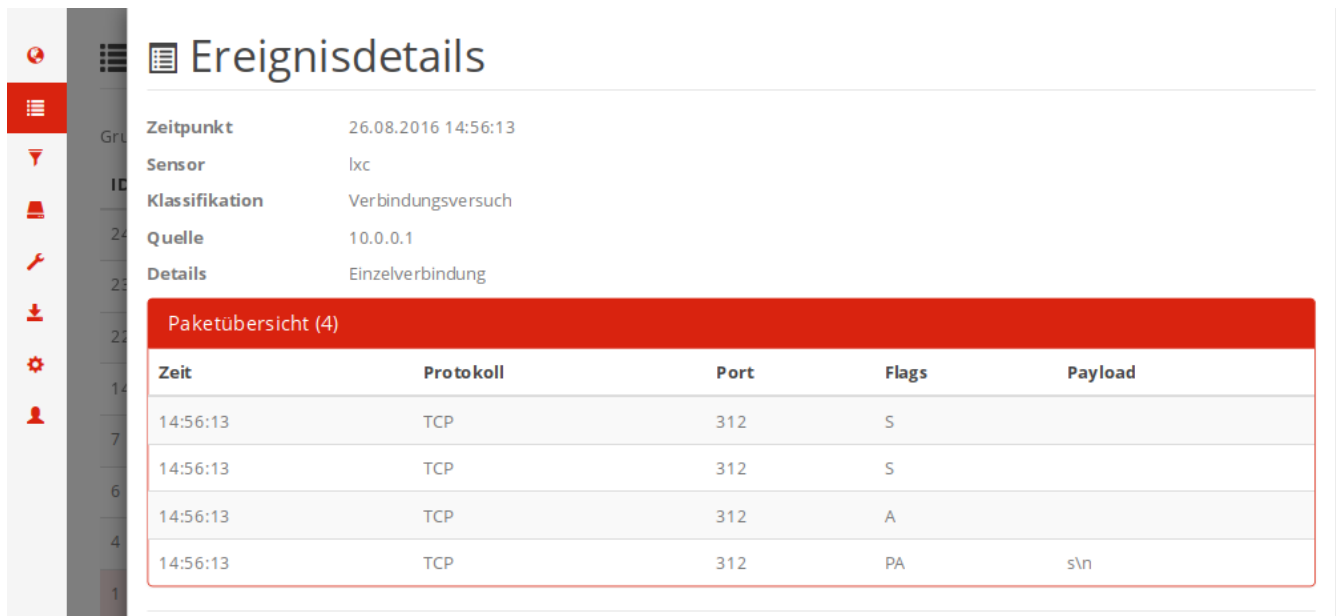
- Nutzer ist mit einem Manager- oder Administratoraccount angemeldet (Kapitel 3)
- Nutzer gehört der Gruppe an, für die die neue Filterregel angelegt werden soll (Kapitel 9.2)

Vorgehensweise Zunächst gilt es, die Ereignisliste auszuwerten und alle nötigen Informationen zum Anlegen des Filters zu sammeln. Dazu wird das Modul *Ereignisse* über die Sidebar aufgerufen und die Sicht auf die Liste, falls nötig, mit Hilfe der Filterfunktionen im oberen Bereich angepasst (siehe auch Kapitel 6). Im Beispiel ist zu sehen, dass wiederholt eine Reihe von Ereignissen von der IP-Adresse



ID	Zeitpunkt	Sensor	Klassifikation	Quelle	Details	Status	Aktionen
24	28.08.2016 10:59:12	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu	[edit] [list] [delete]
23	28.08.2016 10:18:25	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu	[edit] [list] [delete]
22	28.08.2016 10:12:36	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu	[edit] [list] [delete]
14	26.08.2016 14:56:13	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu	[edit] [list] [delete]
7	26.08.2016 14:22:23	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu	[edit] [list] [delete]
6	26.08.2016 14:16:47	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu	[edit] [list] [delete]
4	26.08.2016 14:13:02	lxc	Verbindungsversuch	10.0.0.1	Einzelverbindung	Neu	[edit] [list] [delete]
1	03.08.2016 10:30:54	lxc	Honeypot	10.0.0.1	SSH	Neu	[edit] [list] [delete]

10.0.0.1 verursacht wurden. Mit Klick auf den Button *Details* (1) werden zusätzliche Informationen zu einem gewählten Ereignis sichtbar. In diesen Details kann mit einem Klick auf *Paketübersicht* eine Liste aller vom entfernten Host (und somit potentiellen Angreifer) empfangenen Datenpakete eingesehen werden, die zu diesem Ereignis zugeordnet wurden. Beispielhaft erfahren wir, dass der Sensor mit dem Namen `lxc` das Ereignis aufgezeichnet hat und alle Pakete an den TCP-Port 312 gerichtet waren. Durch Nachforschungen im Produktivnetz finden wir beispielsweise heraus, dass es sich bei dem Rechner mit der IP-Adresse 10.0.0.1 um ein anderes der IT-Sicherheit dienliches System handelt, das mit derartigen Anfragen das Netzwerk scannt, um Hosts zu finden, die Online sind. Nun gilt es, eine Filterregel hinzuzufügen, um solche Vorgänge zukünftig nicht mehr als Vorfall zu melden.



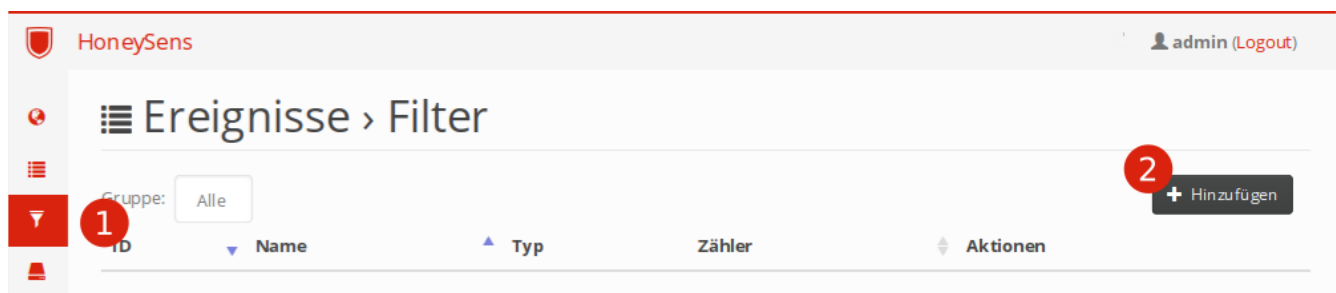
Ereignisdetails

Zeitpunkt: 26.08.2016 14:56:13
 Sensor: lxc
 Klassifikation: Verbindungsversuch
 Quelle: 10.0.0.1
 Details: Einzelverbindung

Paketübersicht (4)

Zeit	Protokoll	Port	Flags	Payload
14:56:13	TCP	312	S	
14:56:13	TCP	312	S	
14:56:13	TCP	312	A	
14:56:13	TCP	312	PA	s\n

Im ersten Schritt wird das Modul für die Ereignisfilter mit einem Klick auf *Filter* in der Sidebar gestartet (1). Es wird eine Liste aller Filter dargestellt, die für die gewählte Gruppe (bzw. standardmäßig alle



HoneySens admin (Logout)

Ereignisse > Filter

Gruppe:

1

ID	Name	Typ	Zähler	Aktionen
----	------	-----	--------	----------

2 + Hinzufügen

Gruppen, denen der aktuelle Benutzer angehört) registriert wurden. Ein Klick auf den Button *Hinzufügen* ruft den Dialog zum Speichern neuer Filter auf, der nachfolgend dargestellt ist (2).

+ Filter hinzufügen

Name: (1)

Typ:

Gruppe: (2)

Filterbedingungen

Attribut	Typ	Wert	Aktionen

+ Hinzufügen (3)

Abbrechen | (4)

Grundsätzlich muss für jeden Filter zunächst ein beliebiger Name (1) vergeben werden, der den Filter innerhalb der Webanwendung identifiziert. Weiterhin ist jeder Filter einer fixen Gruppe zugewiesen (2) und auch nur für die Sensoren und Ereignisse innerhalb dieser Gruppe gültig. Da IP-Adressen in verschiedenen Teilnetzen durchaus mehrmals vergeben sein können, sind global gültige Filterregeln problematisch und werden deshalb nicht unterstützt.

Herzstück des Filtersystems sind die Filterregeln. Jeder Filter setzt sich aus einer Reihe von Regeln zusammen und wird erst auf ein Ereignis angewandt, wenn **ALLE** dieser Regeln zutreffen. Es handelt sich folglich um eine logische UND-Verknüpfung der einzelnen Regeln. Um neue Regeln hinzuzufügen, wird der gleichnamige Button genutzt (3). Anschließend erscheint in der Liste der Filterbedingungen eine neue einzeilige Regel.

+ Filter hinzufügen

Name: ✓

Typ:

Gruppe:

Filterbedingungen

Attribut	Typ	Wert	Aktionen
Quelle (A)	IP-Adresse (B)	10.0.0.1 (C) ✓	✕
Protokoll	IPv4	TCP	✕
Ziel	Port	312 ✓	✕

+ Hinzufügen

Jede individuelle Regel besteht aus einem Attribut (A), einer *Typ* genannten genaueren Spezifizierung (B, je nach Attribut) und einer Wertzuweisung (C). Wenn das gewählte Attribut in den angegebenen

Wertebereich fällt, gilt die Regel als erfüllt. Die obige Abbildung gibt alle drei Regeln an, die unseren Beispiel-Filter umfassen: Pakete von

- der Quell-IP-Adresse 10.0.0.1 (das Sicherheitssystem),
- die das TCP-Protokoll nutzen und
- auf dem Zielport 312 eintreffen

werden fortan vom System auf die Whitelist gesetzt und ignoriert. Wenn alle Regeln ordnungsgemäß eingerichtet sind, kann der Filter mit einem Klick auf *Speichern* auf dem Server registriert und aktiviert werden. Nach dem Anlegen des Filters wird wieder die Filterliste sichtbar, in der ein neuer Eintrag betrachtet werden kann. Der enthält neben dem Namen auch einen Zähler, der immer um eins erhöht wird, wenn ein Ereignis eintrifft, das den Filterbedingungen entspricht und somit verworfen wird. Mit Hilfe des Zählers kann die Wirksamkeit eines Filters leicht überprüft werden (A).

HoneySens admin (Logout)

Ereignisse > Filter

Gruppe: + Hinzufügen

ID	Name	Typ	Zähler	Aktionen
1	Connections from scanner	Whitelist	0 A	✎ ✕