

S(IE)M – Ein Praxisbericht

Matthias Hofherr
atsec information security GmbH
Steinstraße 70
81667 München
matthias.hofherr@atsec.com

Peter Kai Wimmer
atsec information security GmbH
Steinstraße 70
81667 München
peter.wimmer@atsec.com

1 Einführung

Beim Security Event Management (SEM) werden vorrangig sicherheitsrelevante Ereignisse gesammelt und in Realzeit dargestellt. Das Security Information Management (SIM) sammelt Logdaten in einem Netzwerk und analysiert diese auf sicherheitsrelevante Vorkommnisse. SIEM bezeichnet das Bestreben, diese beiden Funktionalitäten zusammenzuführen, auszuwerten und zu berichten (Abbildung 1).

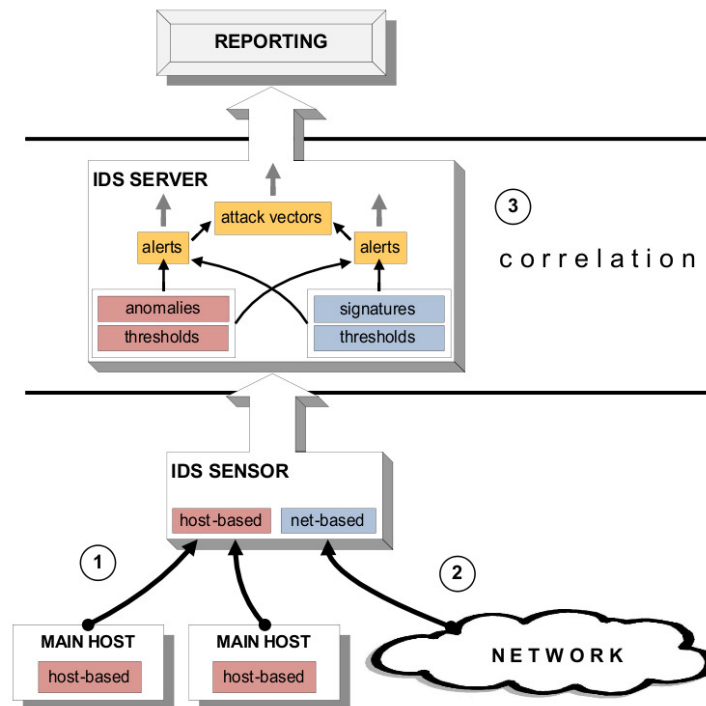


Abbildung 1: SIEM Architektur

2 Überblick

Der Vortrag setzt sich mit dem Thema SIM/SEM/SIEM auseinander und beschreibt zwei verschiedene Ansätze dafür. Der erste Teil betrachtet die Erfahrungen über die letzten Jahre bei der Umsetzung eines SIEM im Enterprise Umfeld mit tausenden heterogener Datenquellen. Teil zwei betrachtet die Implementierung eines SIEM bei der atsec information security GmbH als Inhouse-Projekt mit Open-Source Mitteln. Die Erfahrungen aus Teil zwei sind insbesondere für KMUs und bildende Einrichtungen von Nutzen, da hier die enorm hohen Lizenzkosten von Enterprise-SIEMs entfallen.

3 Enterprise Umgebungen

Ausgangspunkt für diesen Vortrag ist die Beschaffung eines SIEMs für eine Enterprise Umgebung im Rahmen eines Consulting-Projektes. Das Projekt begann mit einer allgemeinen Ausschreibung (RFI) und endete mit einer spezifischen Produktempfehlung für das Umfeld. Der Vortrag soll einen neutralen Überblick über die Fähigkeiten von modernen, kommerziellen SIEM Systemen liefern und die „Lessons Learned“ aus diversen Projekten an das Publikum weitergeben.

3.1 Vorbereitung für die Auswahl des Produkts

Während einer RFI (Request for Information) Phase werden normalerweise verschiedene Hersteller von SIM, SEM und SIEM Lösungen angeschrieben und eine detaillierte Beschreibung ihrer Produkte angefragt. Für diese Beschreibung wurden in vergangenen Projekten mehrere Szenarien definiert, die von den Produkten abgedeckt werden sollten. Des Weiteren wurde eine Anforderungsspezifikation mit üblicherweise über 100 Anforderungen mitgeliefert, die es zu erfüllen galt. Diese Anforderungen umfassen sowohl technische als auch organisatorische Vorgaben. Die Szenarien und die Anforderungen sind bewusst komplex gehalten, um von Anfang an das Teilnehmerfeld auf Kandidaten einzuschränken, die mit Enterprise-Umgebungen vertraut sind.

Als Datenquellen werden meist sowohl proaktive als auch reaktive Komponenten vorgegeben. Ein SIEM muss in der Lage sein, beide Arten von Komponenten zu unterstützen, wobei es sich technisch gesehen um zwei komplett verschiedene Ansätze handelt:

- **Proaktiv:** Hier werden Maßnahmen ergriffen, bevor es zu einem Sicherheitsvorfall kommt. Dies ermöglicht präventive Maßnahmen und verringert damit das Angriffsprofil der IT Umgebung.
- **Reaktiv:** Diese Komponenten unterstützen einen Analysten bei der Ermittlung der Ursache im Falle eines Angriffs.

Eine Kombination dieser beiden Komponenten bringt für ein Unternehmen maximalen Nutzen, aber zu einem hohen Preis (dem Anschaffungspreis).

Beispiele für proaktive Komponenten sind:

- Netzwerkbasierte Vulnerability Scanner
- Hostbasierte System Scanner
- Patch/Update-Systeme
- Network Modelling und Risk Simulation Systeme

An reaktiven Komponenten sollte unterstützt werden:

- Router/Switch logs (diverse Hersteller)
- Firewalls logs (diverse Hersteller)
- AAA Server
- Betriebssystem- und Applikations-Logs (Unix)
- Windows Eventlog
- AV Logs
- RAS Server Logs

- IDS/IPS Events
- Datenbank Security-Logs

Eine Muss-Vorgabe sollte die Integration in ein bestehendes kommerzielles Trouble Ticket System sein, sofern dies bereits existiert. Viele Hersteller von SIEM Produkten sehen zwar vor, dass die Workflows für Security Incident Response innerhalb ihres Produkts modelliert werden. Die wenigsten haben aber Konzepte (oder auch nur Schnittstellen), um bestehende Trouble Ticket Systeme einzubinden. Eine Integration hat den Vorteil, dass nicht für jede Art von Überwachung ein separates Tool zur Verfolgung benötigt wird.

Bedingt durch umfangreiche Vorgaben gibt es in der Regel weniger als zehn Einreichungen, von denen meist auch nur die Hälfte in die engere Auswahl kommt. Primäres Ausscheidungskriterium ist, dass diverse Systeme angeboten werden, die nur mit Produkten eines einzigen oder nur sehr weniger Hersteller zusammenarbeiten. Da aber gerade im Enterprise-Umfeld mannigfaltige (und zum Teil auch hochspezialisierte) Systeme zum Einsatz kommen, sind diese Produkte in der Regel nicht zu empfehlen.

Bei näherer Betrachtung der Angebote aus der Vorauswahl-Phase wird deutlich, dass die Integration von reaktiven und proaktiven Komponenten sehr kostspielig werden kann (insbesondere, wenn viele der Komponenten neu zu beschaffen sind). Hier sollte ein pragmatischer Weg gewählt und nur die unbedingt notwendigen Komponenten eingesetzt werden. Aufwändige Simulationskomponenten sind in der Regel extrem teuer, da es hier auch nicht nur wenige Anbieter auf dem Markt agieren. Unbedingt zu empfehlen ist die Kombination eines IDS / IPS Systems mit einem Vulnerability Scanner. Es sollte darauf geachtet werden, dass der jeweilige Hersteller eine vollständige Integration von IDS/IPS Events zu Vulnerability Scanner Events liefern kann. Eine manuelle Kopplung durch selbst erstellte Korrelationsregeln wäre hier zu aufwändig und kein effektiver Ersatz. Das Mapping von IDS / IPS und Vulnerability ist insofern wichtig, als dass man in der Implementierungsphase des SIEMs nicht beliebig viele allgemeingültige Korrelationsregeln umgesetzt werden können. Hier kann daher am Anfang der Ansatz verfolgt werden, dass nur auf Angriffe korreliert wird, die auch wirklich eine echte Schwachstelle treffen. Damit wird erheblich Zeit bei der Erstellung der Korrelationsregeln eingespart.

Nachdem eine Erstauswahl an verfügbaren Produkten getroffen wurde, werden üblicherweise die Anforderungskriterien verfeinert und mit der Erstellung einer Datenquellen-Matrix begonnen. Diese Datenquellen-Matrix sollte im Detail Informationen über alle im Unternehmen vorhandenen sicherheitsrelevanten Datenquellen beinhalten. Wichtige Informationen sind hier:

- Anzahl pro Gruppe (Windows Server)
- Datenrate pro Quelle (in MB/Tag und Events/Tag)
- Interface (syslog, snmp, proprietär etc.)
- Absicherung, falls vorhanden (stunnel, ssh Tunnel, VPN, proprietär ...)

Ohne diese Informationen ist es einem Hersteller normalerweise nicht möglich, ein passendes Angebot zu erstellen. Bei wirklich großen Projekten empfiehlt sich, schon im Vorfeld nach einer Site License zu fragen.

In der Praxis hat es sich bewährt, schon in dieser Phase ein detailliertes Testkonzept zu erstellen. Dies kann den Anbietern schon im Vorfeld übergeben werden, so dass es gar nicht erst zu Missverständnissen bezüglich der Abnahme kommt. Näheres zum Testkonzept siehe Abschnitt 3.2.

Mit den verfeinerten Anforderungskriterien, dem Testkonzept und der Datenquellen-Matrix folgt als nächster Schritt die RFP (Request for Proposal) Phase. Da in der Regel jeder Anbieter seine Vorschläge in einem anderen Format einreicht, empfiehlt es sich, eine gewichtete Auswertungstabelle vorzubereiten, um bei Eintreffen der Angebote schnell eine Entscheidung treffen zu können. Die Auswertungstabelle sollte von verschiedenen Beteiligten befüllt werden:

- Incident Response Team
- Security Management
- Betrieb
- Einkauf

Da alle Anbieter die gleichen Daten (siehe Datenquellen-Matrix) vorliegen haben, sollten damit auch finanziell vergleichbare Angebote entstehen.

Die am besten geeigneten Anbieter werden in der Regel auf eine Shortlist gesetzt (maximal drei Produkte). Die entsprechenden Produkte auf dieser Shortlist können dann im Rahmen eines Prototyp-Setups getestet werden. Nahezu jeder größere Anbieter für SIEM Lösungen bietet hier kostenlose Proof-of-Concepts an.

3.2 Prototyp Phase

In der Prototyp-Phase werden die verbliebenen Kandidaten der Shortlist jeweils in einer abgeschotteten Testumgebung aufgebaut.

Die Prototyp-Phase gliedert sich idealerweise in zwei Teile:

- Test des Normal-Betriebs: Hier werden reale Test-Datenquellen angeschlossen und die Bedienbarkeit des Systems geprüft.
- Technische Testphase: Hier werden technische Tests aus dem Testkonzept durchgeführt.

Für ein Testkonzept hat es sich bewährt, verschiedene realistische Angriffsszenarien zu definieren. Diese Angriffsszenarien decken jeweils Kombinationen unterschiedlicher Datenquellen ab. Hier sollte darauf geachtet werden, dass nicht nur klassische regelbasierte Erkennung getestet wird, sondern auch die in vielen Produkten beworbene Anomalie-Erkennung. Bei der Anomalie-Erkennung gibt es im Hinblick auf konfigurierbare Parameter große Qualitätsunterschiede. Diese Parameter reichen in der Regel von sehr simplen (nur Anzahl von Events pro Zeiteinheit, mit einem Schwellenwert als Parameter) bis hin zu extrem ausgefeilten Lösungen.

Im Testkonzept sollte auch bedacht werden, sowohl ein agentenbasiertes als auch ein agentenloses Setup zu testen. Wenn möglich sollte soweit als möglich ein agentenloses Setup bevorzugt werden, da dies rein passiv ist und keine Nebenwirkungen auf den überwachten Systemen nach sich zieht. Bei agentenbasierten Lösungen muss in jedem Fall unter realistischen Bedingungen die Verträglichkeit der Agenten auf den Systemen geprüft werden. Insbesondere ist hier auf die CPU Load, die Festplattenauslastung und auf Wechselwirkungen mit dem Betriebssystem oder installierten Applikationen zu achten.

Bei einem Prototyp-Setup sollte auch unbedingt die entstehende Netzwerklast unter realistischen Bedingungen geprüft werden. Werbe-Versprechen mit enormen Kompressionsraten basieren oft auf idealisierten Annahmen. Besonders in Enterprise-Umgebungen fallen häufig außerordentlich große Mengen an Logdaten an, die auch bei einer hohen Kompressionsrate das gesamte Produktivnetz lahmlegen würden. Dies sollte in der Prototyp-Phase erkannt werden, so dass noch über ein dediziertes SIEM / Admin-Netzwerk nachgedacht werden kann (und die entstehenden Kosten in die Kalkulation mit einfließen).

Während der Prototyp-Phase sollte auch geprüft werden, ob das Produkt erweiterte Analyse-Fähigkeiten anbietet. Sehr empfehlenswert ist beispielsweise eine Data-Mining Option, die Muster in bestehenden Daten erkennt und aus diesen Mustern Korrelationsregeln erstellt. Ganz allgemein sollte auch die Handhabung geprüft werden. In der ersten Testphase (Normal-Betrieb) muss die konkrete Nutzung des Produkts für Basisaufgaben im Alltag getestet werden. Hier zeigt sich schnell, dass dies bei manchen Produkten sehr elegant und einfach gelöst ist, während bei anderen Produkten umständliche Prozeduren selbst für kleinere Aufgaben nötig sind.

Für die Prototyp-Phase hat es sich bewährt, eine Testsuite zu erstellen. Hierzu werden mit Hilfe einer Skript-Sprache verschiedene Angriffsszenarien aus dem Testkonzept simuliert. Als Vorlage können hier Originaldaten herangezogen werden. Es muss allerdings darauf geachtet werden, dass die Umsetzung sehr sauber erfolgt, da bei einem Fehler die Hersteller keine Möglichkeit haben, das Szenario korrekt umzusetzen. Üblicherweise werden die Tests auf einer dedizierten Testmaschine mit virtuellen Interfaces gestartet, so dass jedes Interface eine eigene Datenquelle simuliert. Dies vermeidet Fehler bei manchen Produkten, die Probleme mit der Einbindung mehrerer Datenquellen mit identischen IP-Adressen haben.

Für die Testphase sollte beachtet werden, dass in der Regel keine Möglichkeit für Lasttests gegeben sind. Meist stehen Hardware oder Support-Systeme (z.B. eine externe Datenbank) nicht in ausreichender Dimension für realistische Tests zur Verfügung. Hier empfiehlt es sich, über Referenzen festzustellen, ob es bereits Installationen des Produkts mit einem ähnlichen oder größeren Umfang gibt. Sollte die eigene Umsetzung unfreiwilligerweise als „Bleeding-Edge-Innovator“ erfolgen, dann empfiehlt es sich, dies im Vertragswerk später entsprechend abzusichern.

3.3 Implementierung

Hat man sich in der Prototyp-Phase für ein Produkt entschieden, dann sollte noch vor der Implementierung in jedem Fall sichergestellt werden, dass im Haus die nötigen Ressourcen für Security Incident Handling zur Verfügung stehen. Ohne ein Team, das die Alarme des SIEMs tatsächlich auswertet, ist das investierte Geld nutzlos angelegt. Dieses Team muss auch die nötige Zeit zur Verfügung haben, das SIEM zu warten und die Regelsätze für die Korrelation anzupassen.

Nach einer Entscheidung für ein Produkt sollte ein detaillierter Rollout-Plan erstellt werden. Dabei ist zu beachten, dass nicht alle Datenquellen auf einmal angeschlossen werden. Ein schrittweiser Anschluss ist hier empfehlenswert, da etwaige Lastprobleme unter Umständen erst an dieser Stelle im Projektplan erkannt werden.

Abschließend bleibt anzumerken, dass die Qualität eines SIEMs von der Qualität der eingespeisten Daten abhängt. Daher ist darauf zu achten, dass für jeden Datenquellen-Typ eine Logging-Richtlinie erstellt wird (falls noch nicht vorhanden), welche die minimale benötigten Informationen festlegt.

4 Nicht-kommerziell

Ziel eines internen Projektes der *atsec information security GmbH* ist die Evaluierung von Open-Source Werkzeugen für den Aufbau eines SIEMs für kleine und mittlere Unternehmen (KMUs). Die Fragestellung ist hierbei, ob für ein avisiertes Budget von etwa 10.000,- Euro ein effektives und aussagekräftiges SIEM implementierbar ist, welches sich ohne hohen Wartungsaufwand und somit ohne hohe Folgekosten betreiben läßt.

Gleichzeitig werden Metriken implementiert, welche die Forderung verschiedener Standards zur Informationssicherheit nach Meßbarkeit von KPIs (Key Performance Indikatoren) erfüllen. Hieraus läßt sich auch die Qualität der Lösung ablesen.

4.1 Verwendete Werkzeuge

Sowohl Host- als auch Netz-basierte Informationsquellen sollen als Quellen sicherheitsrelevanter Ereignisse dienen. Zur Analyse des Netzverkehrs werden Sensoren in relevanten Subnetzen implementiert; die Ereignisse auf Hosts werden von den Diensten selbst sowie vom Kernel via syslog geliefert.

4.1.1 Information über sicherheitsrelevante Ereignisse

Aus der Vielzahl frei verfügbarer Werkzeuge (siehe u.a. die Liste auf <http://sectools.org/>) wurden folgende Netz-basierte Komponenten (Tabelle 1) ausgewählt:

Werkzeug	Beschreibung	Verwendung
Snort	Netzwerk IDS	Erkennung Netz-basierter Angriffe
p0f	Passiver OS-Fingerprint	Betriebssysteme von Hosts
arpwatch	Zuordnung ARP- zu IP-Adressen	Aufspüren neuer / geänderter Hosts in einem Netzwerk
pads	Passive Asset Detection System	Erkennung aktiver Netzdienste

Tabelle 1: Werkzeuge für Netzwerkanalyse

Auf Hosts werden die syslog-Meldungen wichtiger Dienste ausgewertet (Auswahl in Tabelle 2):

Werkzeug	Beschreibung	Verwendung	
Standard UNIX (via syslog)	ipmon	Protokollierung (gefilterter) Pakete	
	shhd	Informationen über erfolgreiche / fehlgeschlagene Login-Versuche u.ä.	
	samba		File Service
	openvpn		Virtual Private Network
Kernel	Kritische Kernel-Meldungen	Auswertung ungewöhnlicher Systemzustände	

Tabelle 2: Quellen für die hostbasierte Analyse

Weitere Werkzeuge lassen sich nach Bedarf zusätzlich einbinden.

4.1.2 Betriebsunterstützende Werkzeuge

Zur Verarbeitung der sicherheitsrelevanten Ereignisse und der Sicherstellung eines kontinuierlichen Betriebes werden zusätzlich die Werkzeuge in Tabelle 3 verwendet.

Werkzeug	Beschreibung	Verwendung
SEC	„Simple Event Correlator“	Korrelation (siehe 4.3.2)
syslog-ng	(zentraler) Log Host	Grobfilterung, Weiterleitung von syslog-Meldungen
restartd	Neustart fehlerhafter Prozesse	für die Werkzeuge aus 4.1.1

Tabelle 3: Werkzeuge zur Betriebsunterstützung

4.2 Architektur

Ein zentraler IDS Server wertet verdächtige Ereignisse von den Sensoren an den verschiedenen Standorten aus (siehe Abbildung 1). Diese Sensoren wiederum sammeln Netz- und Host-basierte sicherheitsrelevante Ereignisse; ein Sensor dient dabei als Log-Host für relevante syslog-Meldungen von überwachten kritischen Systemen.

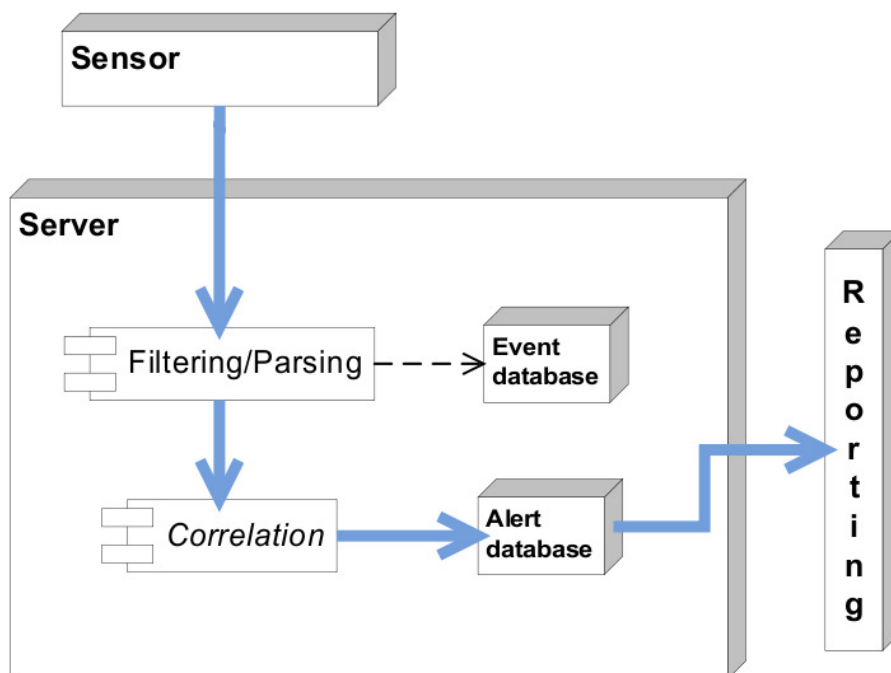


Abbildung 2: Architektur des SIEM

Ein hohes Maß an Modularität ist gewährleistet durch die Trennung von Informationsbeschaffung, Filterung und Analyse, sowie durch die Trennung von Ereignis-Erkennung und Alarm-Bearbeitung während der Analyse (siehe 4.3).

4.2.1 Informationsfluß

Diese Werkzeuge schicken Ereignisse jeweils an einen lokalen syslog-ng, der sie an den zentralen syslog-ng auf dem IDS Server weiterleitet. An den größeren Standorten sendet Snort seine Ereignisse zusätzlich an eine lokale Datenbank auf dem IDS-Sensor; die Snort-

Sensoren an kleineren Standorten haben keine lokale Datenbank, sondern senden ihre Ereignisse an die Datenbank des zentralen IDS Servers.

Aus Redundanzgründen und für eine nachträgliche Analyse werden alle anderen Ereignisse ebenfalls in der Server-Datenbank abgelegt.

4.3 Auswertung

Die Auswertung der Meldungen erfolgt durch Filterung (4.3.1) und Korrelation (4.3.2).

4.3.1 Filterung

Die erste Instanz von SEC nimmt alle syslog-ng Ereignisse entgegen, filtert unwichtige Ereignisse heraus (4.3.1.1) und wandelt verdächtige Ereignisse in ein einheitliches Format um (4.3.1.2).

4.3.1.1 Filter und Kategorisierung

Zur Zeit werden die in Tabelle 4 aufgeführten Ereignisse als verdächtig angesehen und mit einem entsprechenden Label kategorisiert. Alle anderen Meldungen werden verworfen (*Reduktion*).

Label	Ereignisse
STATUS INFORMATION	Standard p0f events
	Arpwatch 'new activity' events
SUSPICIOUS BEHAVIOUR	Arpwatch 'new station' events
	Arpwatch 'flip flop' events
	Arpwatch 'changed ethernet address' events
RECONNAISSANCE / BAD TRAFFIC	ipmon 'Deny packet' events
	Austin firewall events
	Snort 'Attempted Information Leak' events
	Snort 'Information Leak' events
	Snort 'Large Scale Information Leak' events
ACCESS ATTEMPT	Snort 'Potentially Bad Traffic' events
	Sshd 'Invalid User' events
	Sshd 'Failed password' events
	Sudo 'authentication failure' events
	su 'failed' events
	Snort 'Attempted User Privilege Gain' events
POSSIBLE COMPROMISE	Snort 'Attempted Administrator Privilege Gain' events
	Pads events
	'restartd does some restarting' events
	kill events

Tabelle 4: Filterung

4.3.1.2 Einheitliches Format

Die als verdächtig eingestuft Ereignisse werden in folgendes Format umgewandelt:

<label>, <priority>, <sensor>, <generator>, <source IP>, <source port>, <destination IP>, <destination port>, <event ID>, <original message>

Die einzelnen Parameter haben dabei folgenden Inhalt (Tabelle 5):

Parameter	Verwendung	Inhalte
<label>	hauptsächlich für die folgende Korrelation (siehe 4.3.2)	INFO: status information Ereignisse von Werkzeugen wie p0f oder pads SSP_BHV: suspicious behaviour Verdächtige Ereignisse von Clients ohne klares Ziel; Beispiel: Änderung der MAC-Adresse RECON: reconnaissance Netzverkehr, der evtl. Informationen über das interne Netz offenlegt Beispiele: deny-Pakete von ipmon, Broadcasts ACCESS: access attempt Fehlgeschlagene Anmeldungen; bekannte Exploits COMPROM: possible compromise Ereignisse, die auf eine Kompromittierung des Systems hindeuten Beispiele: kritische Kernel-Meldungen, neue Dienste
<priority>	Korrelation	Snort Prioritäten, ansonsten 0
<sensor>	-	IP Adresse des Sensors
<generator>	-	Name des Werkzeugs, von dem das Ereignis stammt
<source IP>	Korrelation	Verbindungsdetails (nicht für alle Ereignisse vollständig verfügbar)
<source port>		
<destination IP>		
<destination port>		
<event ID>	Korrelation, Rückverfolgung	Eineindeutige ID
<original message>	-	Die vollständige syslog-Meldung

Tabelle 5: Einheitliches Format

4.3.2 Korrelation

Die Korrelation ist das Herzstück des SIEM. Gemeinsamkeiten von Ereignissen werden nach *Quelle, Ziel* sowie *Art des Ereignisses* erkannt (siehe Tabelle 6) und diese gegebenenfalls zu einem eigenen Ereignis („Alarm“) zusammengefaßt. Dabei kommt ein zustandsbasiertes Modell zum Einsatz: die Kritikalität eines Alarms durch die Anzahl der beteiligten Ereignisse sowie deren Kritikalität erhöht, bzw. sinkt auch wieder, sobald die zugehörigen Ereignisse ausbleiben.

Quelle	Ziel	Anzahl, Art des Ereignisses	Beschreibung
1	1	Zugriff auf verschiedene Ports	Portscan
1	viele	Zugriff auf einen Port	Wurm versucht, sich weiter zu verbreiten
viele	1	Viele Zugriff auf einen Port	DDOS
1	1	Viele fehlgeschlagene Anmeldeversuche	Brute Force Versuch, Paßwörter zu erraten

Tabelle 6: Beispiele für Korrelation

Zur Korrelation der Ereignisse wird der „Simple Event Correlator“ SEC (<http://kodu.neti.ee/~risto/sec/>) verwendet.

Folgende Regeln sind bis jetzt implementiert, zur Erkennung von:

- Vertikalen und horizontalen Port Scans
- Lokalen und entfernten fehlgeschlagenen Anmeldungen
- Allgemeine Aufklärung (Reconnaissance), Zugriffe oder Kompromittierungen

Außerdem:

- Angriffsvektoren (in der Testphase)
- Historische Analyse (in der Planung)

4.4 Sicherheits-Metriken

Informationssicherheits-Management-Systeme, etwa nach ISO 27001, erfordern die Implementierung eines kontinuierlichen Verbesserungsprozeß (KVP) bzw. Plan-Do-Check-Act Zyklus (PDCA). Ein solcher Prozeß benötigt Informationen sowohl über den aktuellen als auch den bisherigen Zustand der Sicherheit. Somit sind Kennzahlen (Key Performance Indicator, KPI) zur Bewertung der verschiedenen Aspekte der Sicherheit passend zum jeweiligen Umfeld zu definieren.

Die Analyse von Trends anhand dieser Metriken ist ein adäquates Mittel zur Steuerung von gezielten Investitionen in die IT-Infrastruktur oder auch Einführung organisatorischer Maßnahmen zur Verbesserung der Informationssicherheit. Gleichzeitig wird die Effizienz bisheriger Maßnahmen reflektiert.

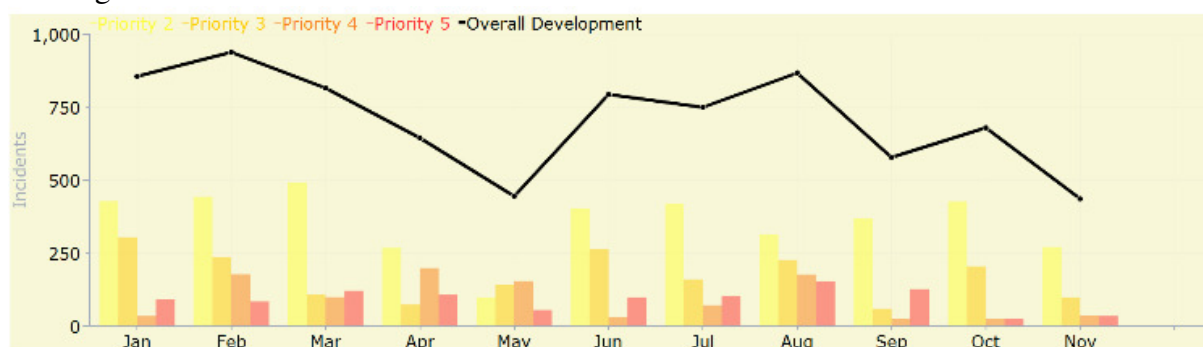


Abbildung 3: Metriken zur Sicherheit

Abbildung 3 zeigt beispielhaft eine hohe Anzahl von sicherheitsrelevanten Vorkommnissen zu Beginn des Beobachtungszeitraumes, die durch die Einführung von zusätzlichen Sicherheitsmaßnahmen (Feb. bis Mai) reduziert werden konnten, danach durch neuartige Bedrohungen allerdings auch wieder anstiegen.

Solche Diagramme werden durch spezielle statistische Auswertungen über die Datenbank mit den historischen Ereignissen bzw. Alarmen umgesetzt. Ein eigenes Web-Interface liefert hierzu sowohl interaktive Charts als auch Berichte als PDF (Reporting).

4.5 Fazit und Ausblick

Im Testbetrieb bei atsec und in einem universitären Netzwerk haben sowohl die eingesetzten Open-Source Applikationen als auch die entwickelte Korrelation ihre Tauglichkeit unter Beweis gestellt. Die Modularität der entwickelten Lösung erlaubt den bedarfsgerechten Ausbau und den Einsatz in unterschiedlichen Netzarchitekturen.

Zur Zeit werden einige interessante Erweiterungen untersucht:

- **Asset Liste**
Eine werkzeuggestützte Erstellung und Pflege der u.a. von ISO 27001 geforderten IT-Asset Liste. Zum Einsatz kommen hierbei Programme wie p0f und pads (siehe 4.1.1), die Informationen zu Betriebssystem, Netzwerkkonfiguration und angebotene Dienste beisteuern. Hieraus lassen sich außerdem (unbeabsichtigte oder mutwillige) Änderungen der Konfiguration als auch Trends bzgl. der Server und Clients ablesen.
- **Proaktive Warnung vor Schwachstellen**
Aus Meldungen von Schwachstellen-Scannern (z.B. Nessus), Paßwort-Knackern (John the Ripper) oder Integritäts-Prüfern (Tripwire, Samhain) werden Alarme für die betroffenen Systeme generiert
Eine **Datenbank aktueller Schwachstellen** im System liefert eine weitere wichtige Metrik zur Beurteilung der IT-Sicherheit
- **Anbindung an Trouble-Ticket System**
Bestimmte Alarme (wie etwa „Schwachstelle gefunden“) lassen sich sinnvollerweise als Ticket in ein Trouble Ticket System überführen. Dies würde die Behebung solcher Probleme fördern und eine nachvollziehbare Dokumentation gewährleisten.
- **Erweiterte Korrelation**
Ein Abgleich eines festgestellten Angriffes mit der oben angedachten Datenbank aktueller Schwachstellen sowie der Asset-Liste reduziert die Anzahl kritischer Alarme. Die Korrelation berücksichtigt hierbei, ob das angegriffene System überhaupt eine korrespondierende Schwachstelle aufweist oder das „passende“ Betriebssystem verwendet.

5Literaturverzeichnis

- [1] Schlamp, Johann (2008), *Entwurf und Implementierung eines metrikbasierten Reporting-Systems für IT-Sicherheit*, Systementwicklungsprojekt, Fakultät für Informatik der Technischen Universität München, <http://www.nm.ifi.lmu.de/pub/Fopras/schl08/>
- [2] *UNIX syslog-Protokoll, RFC 3164*, <http://www.ietf.org/rfc/rfc3164.txt>
- [3] *Simple Event Correlator (SEC)*, <http://kodu.neti.ee/~risto/sec/>
- [4] *Top 100 Network Security Tools*, <http://www.sectools.org/>
- [5] Snort, <http://www.snort.org/>
- [6] p0f, <http://lcamtuf.coredump.cx/p0f.shtml>
- [7] arpwatch, <ftp://ftp.ee.lbl.gov/arpwatch.tar.gz>
- [8]