SelfLinux-0.13.0

Intrusion Detection Systeme - IDS

Autor: Florian Frank (*florian.frank@pingos.org*)
Autor: Gabriel Welsche (*gabriel.welsche@web.de*)
Autor: Alexis Hildebrandt
Autor: Mathias Meyer
Formatierung: Florian Frank (*florian.frank@pingos.org*)

Intrusion Detection (ID) oder zu deutsch **Einbruchserkennung** ist noch ein sehr junges Gebiet der Netzwerksicherheit. Dieser Artikel soll die theoretischen Grundlagen und praktischen Möglichkeiten der Einbruchserkennung beleuchten. Obwohl Intrusion Detection Systeme (IDS) oftmals noch für **unreif** gehalten werden, so ist ihre Rolle als Komponente in einer Sicherheitsarchitektur sehr bedeutsam.

Inhaltsverzeichnis

1 Einführung in das Gebiet der "Intrusion Detection"

Mit der wachsenden Verbreitung des Internets wächst auch der Bedarf an Sicherheit - sowohl im privaten als auch im geschäftlichen Bereich. Vornehmlich Firmennetzwerke sind oftmals Angriffsversuchen ausgesetzt. Um den Schaden abzuwenden bzw. zu begrenzen, müssen diese Angriffe erkannt und entsprechende Gegenmaßnahmen eingeleitet werden.

Das Gebiet der **Intrusion Detection** (ID - zu dt. Einbruchserkennung) befasst sich mit der Entwicklung von Methoden zur frühzeitigen Angriffserkennung. Ein IDS kann demnach als **Alarmanlage** angesehen werden. Durchbricht beispielsweise ein Angreifer die Sicherheitsvorkehrungen (Firewall), so soll das IDS Alarm auslösen. Dies kann eine Mitteilung an den Administrator (E-Mail, SMS, Pager, etc.) oder aber aktive Gegenmaßnahmen, wie das Schließen von Ports, das Beenden angegriffener Dienste, etc. beinhalten. Des Weiteren sollte der Administrator genaue Informationen über den Angriff selbst erhalten, damit eventuelle geeignete und effektive Gegenmaßnahmen möglichst schnell eingeleitet werden können.

Doch woraus besteht ein Intrusion Detection System? Welche Möglichkeiten stellen die verschiedenen Systeme zur Verfügung? Wie setzt man es erfolgreich ein und welche Instrumente stehen unter Linux zur Verfügung?

1.1 Begriffsklärung und Einordnung der verschiedenen Systeme

Um ein Intrusion Detection System konkretisieren zu können, ist zunächst die Frage zu klären, durch was **Eindringen** (Intrusion) überhaupt gekennzeichnet ist. *Heberlein*, *Levitt* und *Mukherjee* von der University of California, und *Davis* haben sie 1991 wie folgt definiert:

> "... eine Menge von Handlungen, deren Ziel es ist, die Integrität, die Verfügbarkeit oder die Vertraulichkeit eines Betriebsmittels zu kompromittieren."

Daraus ergibt sich folgende Deutung für Intrusion Detection:

> "Intrusion Detection ist die Gesamtheit aller Methoden, um gegen ein System oder Netzwerk gerichtete Aktionen herauszufinden"

Im Laufe der Zeit bildeten sich vier Arten an ID-Systemen heraus:

* Logfile-Analysatoren,
* Host basierte IDS (HIDS),
* Netzwerk basierte IDS (NIDS),
* Deception-Systeme (auch bekannt als Honeypots).

Logfile-Analysatoren untersuchten die System-Protokolldateien auf ungewöhnliche Ereignisse (Mustererkennung). Durch die fehlende **Echtzeitfähigkeit** sind sie nicht in der Lage, heute übliche Angriffe z. B. durch Rootkits zu erkennen - geschweige denn - ihnen entgegenzuwirken.

Bei Host basierten Intrusion Detection Systemen (HIDS) läuft ein Teil der Überwachungssoftware (nämlich die Ereigniskomponente auf dem zu überwachenden Rechner (Host), d. h. dass die zu analysierenden Daten von diesem Host stammen. Angriffsanalyse, Protokollierung und eventuelle Gegenmaßnahmen sind nicht zwangsweise auf einen Rechner beschränkt. Zur Gruppe der HIDS zählen die so genannten **System-Integritäts-Verifikatoren** (SIV), zu denen z. B. auch `tripwire` und samhain gehören.

Netzwerkbasierte IDS überwachen den Netzwerkverkehr (Netzwerkpakete), als Vertreter ist `snort` zu nennen. Die meisten Netzwerk-IDS arbeiten im promiscious-Modus.

Honeypots (Honigtöpfe) geben fälschlicherweise vor, wichtige Daten vorzuhalten und lenken damit von den tatsächlich sicherheitskritischen Systemen ab. Bei einem Angriff protokollieren sie alle Aktivitäten des Angreifers mit dem Ziel, diesen möglichst frühzeitig überführen zu können und zwar ohne sensitive Daten preiszugeben.

1.2 Gründe für Intrusion Detection Systeme

Die Motivation, ein Netzwerk oder auch nur einen Einzelrechner mit einem Intrusion Detection System abzusichern, liegt heutzutage auf der Hand. Kaum ein Tag vergeht, an dem man nicht von neuen Crackversuchen oder Einbrüchen in Firmennetzwerken hört. So manche Kreditkartennummer hat durch Einbruchsversuche schon den Weg in die Finger böswilliger krimineller Individuen gefunden.

Informationen des CERT zufolge stiegen in den letzten Jahren die Anzahl der Angriffe und die damit verbundenen Kosten exponentiell an. Dieser Tendenz konnte bisher noch nichts entgegengesetzt werden, auch so genannte Intrusion Prevention Systeme (zu dt. **Einbruchsverhinderungssysteme**) verfehlten oftmals ihren Zweck.

Bedrohungen können sehr vielfältig Art und Weise auftreten. Der Angreifer kann zum Beispiel über Fehler in der Implementierung des TCP/IP-Stacks Zugriff auf das System oder dessen Ressourcen erhalten. Ein Profi findet anhand einer Untersuchung des TCP/IP-Fingerprints (nmap heraus, um welches System es sich handelt, kann durch diese Information ganz gezielt Schwachstellen suchen und nutzt diese letztendlich für einen Angriff aus.

Schwachstellen lassen sich aber auch oftmals in der auf dem System vorhandenen Software finden - meist in Form von Pufferüberläufen (Buffer Overflows), durch die ein normaler Benutzer u. U. sogar root-Rechte erlangen kann. Besonders negativ aufgefallen sind der DNS-Daemon BIND, welcher schon seit Jahrzehnten unter Unix und seinen Derivaten markführend ist, der FTP-Server WuFTP und natürlich der **Klassiker**, der Microsoft Internet Information Server. Durch die ersten beiden Programme gelangten die ersten Würmer in die Unix-Welt. Mittlerweile gibt es sogar einen Wurm, der eine Kombination aus Sicherheitslöchern im Microsoft IIS und BIND ausnutzt.

Fälschlicherweise wird oft angenommen, dass Einbruchsversuche grundsätzlich nur in Firmen-Netzwerken oder in großen Organisationen geschehen. Tatsächlich ist jeder private Internetnutzer den Gefahren eines Angriffs ausgesetzt. Aber eine IP-Adresse, von der man weiß, dass die dahinter steckenden Rechnersysteme häufig wechseln, ist nicht halb so interessant wie ein System, welches dauerhaften Kontakt zum Internet hat oder gar mehrere Systeme in einem Netzwerk, die für einen der gefürchteten Distributed Denial of Service-Attacken genutzt werden könnten. Ein Rechner, dessen Internet-Anbindung mittels Flatrate-DSL zustande kommt, ist weitaus mehr gefährdet als einer, der ISDN oder ältere Modems nur sporadisch nutzt.

Letzten Endes ist eine Kette immer nur so stark wie ihr schwächstes Glied - der Benutzer des Systems. Oftmals sind Einbruchsversuche auf falsch gewählte Passwörter zurückzuführen. Hat ein Einbrecher erst einmal einen Zugang zum System, ist es ein Leichtes, zu noch höheren oder gar den absoluten, den root-Rechten, auf einem Rechner zu gelangen. Neben so genannten **Brute-Force-Angriffen** kann der Angreifer natürlich auch mittels **Social Engineering** versuchen, an das Passwort des Benutzers zu gelangen. Man hört immer wieder von Fällen, in denen dreiste Personen unter einem Vorwand oder Vorgabe einer falschen Identität versucht haben, einen Benutzer zur Herausgabe seines Passwortes zu bewegen. Ein IDS kann natürlich einen solchen (Passwort)-Angriff nicht verhindern. Es lassen sich höchstens ungewöhnliche und von diesem Login ausgehende Aktivitäten erkennen, aber dann ist es leider oft zu spät.

Ohne Intrusion Detection System besteht keine Möglichkeit herauszufinden, wie lange ein Eindringling unbemerkt blieb, wie er seinen Angriff ausführte und welcher Schaden dabei entstand. Zusammenfassend lassen sich die Ziele von ID-Systemen in folgenden Punkten ausdrücken:

* im Falle eines Angriffs Benachrichtigung der Verantwortlichen (Administrator, Sicherheitsbeauftragter) oder aktive Gegenmaßnahmen,
* juristische Verwertbarkeit der gesammelten Erkenntnisse,
* Erkennung von Datenverlusten,
* Schutz vor zukünftigen Angriffen durch Auswertung der gesammelten Erkenntnisse bei einem (simulierten) Einbruch.

1.3 Anzeichen für ein "Eindringen"

Ein Intrusion Detection System deutet Unregelmäßigkeiten im System als Anzeichen eines Einbruchs. Man unterscheidet:

Systembezogene Anzeichen:

* ungewöhnliche Login - Aktivitäten (z. B. nächtliches Anmelden, Login auf lange unbenutzte Benutzerkonten, neue ungewöhnliche Benutzerkonten, etc.),
* ungewöhnliche Systemprozesse oder Kernel-Module,
* abnormal hohe CPU-/Speicherauslastung, ungewöhnliches Systemverhalten (z. B. Systemabstürze, Neustarts),
* Zeitlücken bzw. auffallende Zeitangeben in der Protokollierung (Logfiles).

Anzeichen im Dateisystem:

* veränderte Benutzer/Gruppenzugehörigkeit bzw. veränderte Zugriffsrechte insbesondere der Protokoll-Dateien (Logfiles),
* veränderte Dateigrößen, Dateiinhalte (Systemsoftware, Konfigurationsdateien, Logfiles, ...),
* neue, ungewöhnliche Dateien/Programme,
* neue SUID-, SGID-Dateien,
* nicht mehr vorhandene (gelöschte) Dateien.

Netzwerkspezifische Anzeichen:

* Verbindungen von ungewöhnlichen Standorten (IP-Adressen) zu ungewöhnlichen Zeiten (z. B. nachts),
* erhöhter Netzwerkverkehr (im Extremfall DoS),
* Anfragen auf unbenutzten/geschlossenen Ports,
* Anfragen, die auf einen Port-Scan hindeuten.

Die in den ersten beiden Gruppen aufgeführten Anhaltspunkte werden von HIDS insbesondere den System-Integritätsverifikatoren behandelt, die letzte Gruppe hingegen wird von Netzwerk-IDS identifiziert.

2 Aufbau und Funktionalität eines IDS

Im Allgemeinen erledigt ein MD-IDS seine Arbeit in drei Schritten. In den folgenden drei Abschnitten werden die diese Schritte erläutert. Daran anschließend wird der prinzipielle Aufbau eines IDS besprochen.

2.1 Datensammlung

Im ersten Schritt werden Daten gesammelt. Im Falle eines Netzwerk-IDS handelt es sich hierbei um Pakete, die das Netzwerk passieren. Bei integritätsüberprüfenden Systemen (SIVs) hingegen besteht die Datensammlung aus charakteristischen Systemmerkmalen (Datei-Hash-Werten, Dateigrößen, Zugriffsrechte, etc.), die in Form von Signaturen abgelegt werden. Weitere wichtige Daten werden von solchen Komponenten geliefert, die die

Vergabe von Betriebsmitteln durch das Betriebssystem überwachen (CPU, Speicher-Auslastung, aktive Netzwerkverbindungen, etc.). Protokolldaten (Logfiles) sind ebenfalls essenzielle Datenquellen.

Besonders wichtig ist die Vertrauenswürdigkeit gegenüber der Datenbezugsquelle. Informationen, die mitgehört oder gar manipuliert werden können, sind nicht nur nutzlos sondern äußerst gefährlich! (z. B. Logfiles, in denen gelöscht werden kann, ...)

2.2 Datenanalyse - Anomalieerkennung vs. signaturbasierter Missbrauchserkennung

Zwei grundlegende Techniken zur Datenanalyse wurden in den letzten Jahren entwickelt:

* Anomalieerkennung (Anomaly Detection -AD-IDS),
* Missbrauchserkennung (Missuse Detection -MD-IDS).

Bei der Missbrauchserkennung wird anhand vordefinierter Muster versucht, Einbrüche zu erkennen. Die Anomalieerkennung stellt Abweichungen zum Normalbetrieb fest. Auch wenn AD-IDS kaum in der Praxis zu finden sind, so besitzen sie einen sehr wichtigen Vorteil gegenüber MD-IDS, welcher darin besteht **unbekannte** Angriffsmuster zu erkennen.

Zur Anomalieerkennung werden zwei Techniken eingesetzt, die Erste basiert auf Methoden der künstlichen Intelligenz. Dem AD-IDS wird in einer Trainingsphase der Normalzustand des Systems angelernt. Das Ergebnis des Trainings ist ein Systemprofil, welches als Vergleichsmuster zur Erkennung von Abweichungen dient. Die Fähigkeit, auch unbekannte Angriffsszenarien zu entdecken, setzt jedoch sehr gute Statistikkenntnisse und umfangreiches Training voraus, um ein solches System auch entsprechend nutzen zu können. Eine zweite Herangehensweise der Anomalieerkennung verfolgt einen logischen Ansatz. Entscheidend ist die zeitliche und logische Abfolge von Ereignissen. Beobachtet das System den Anfang einer Ereignisfolge, erwartet es, dass auch der Rest dieser Ereignisfolge abläuft. Passiert dies nicht, schlägt das System Alarm.

Missbrauchserkennungssysteme (MD-IDS) lassen sich am Besten mit einem Virenscanner vergleichen. Bereits bekannte Angriffsmuster liegen in Form von Signaturen vor, die - analog zum Virenscanner - möglichst immer auf einem aktuellen Stand sein sollten. Im Internet sind Unmengen an Angriffssignaturen zu finden. Die Voraussetzung für die Missbrauchserkennung ist eine für den zu identifizierenden Angriff passende Signatur und - selbstverständlich - dass das MD-IDS diese Signatur berücksichtigt.

Der letzte Abschnitt führte den Begriff der Signatur (als Fingerabdruck eines Angriffs) ein. Zum Beispiel erzeugt ein Angriff über ein Netzwerk bestimmte Pakete, die sich entweder durch einen String auf Applikationsebene, wie z. B. /cgi-bin/phf für ein CGI-Probe, durch eine bestimmte Konstellation gesetzter TCP-Flags, wie z. B. alle Flags bei XMas-Tree-Scan (siehe nmap) oder ähnliche Merkmale auszeichnen. So kommt die Signatur zustande, die ein Angriffsmuster repräsentiert.

2.3 Datenvisualisierung und eventuelle Einleitung von Gegenmaßnahmen

Die Darstellung des Ergebnisses der Analyse geschieht in Abhängigkeit der verwendeten Erkennungstechnik. Die zu visualisierenden Informationen lassen sich in folgende Gruppen einteilen:

* Art des potentiellen Angriffs/Unregelmäßigkeit,
* Daten zur Identität des Angreifers (IP, ...),
* Zeitangaben (vermutete Dauer des Angriffs, ...),
* Daten zu angegriffenen Diensten,
* Schadensanalyse (veränderte, gelöschte Daten),
* usw.

Mittels statistischer Graphiken lassen sich Informationen, die über einen längeren Zeitraum gesammelt wurden, benutzerfreundlich aufbereiten. Einige ID-Monitore bieten außerdem Erklärungen zum Grund der Angriffsmeldung und zeigen u. U. sogar potentielle Schwachstellen, die vom Angreifer ausgenutzt wurden.

Die Ausgabe der Ergebnisse beschränkt sich natürlich nicht auf die Aufbereitung der Daten und einer entsprechenden passiven Anzeige (z. B. Web-Interface wie ACID). Notwendig sind auch aktive Benachrichtigungsmethoden wie z. B. E-Mail- oder SMS-Versand.

Unter Umständen sind ID-Systeme in der Lage, anhand der analysierten Daten entsprechende Gegenmaßnahmen einzuleiten. Man teilt IDS anhand ihrer Reaktionsschnelligkeit in folgende zwei Gruppen ein:

* reaktionär
* unmittelbar wirkend
  * automatische Aktionen
  * (halb)automatisch - Interaktion mit Administrator
  * manuell - dem Administrator werden Vorschläge unterbreitet.

Reaktionäre Systeme werden heute kaum weiterentwickelt, der Fokus liegt eindeutig auf unmittelbar wirkenden IDS. Trotz des Vorteils einer sehr schnellen Angriffserkennung sei auf einige Nachteile dieser **Live**-Systeme hingewiesen:

* extensiver Ressourcenverbrauch und dadurch hohe Kosten,
* manipulierbar und angreifbar (durch gezielte Attacken), im worst case Nicht-Verfügbarkeit des IDS,
* untersucht meist nur Paketkopf (nicht Paketinhalt),
* viele falsch-positive Alarme (siehe BURGAR-Alarms als möglichen Ausweg, Idee dieser Technik ist das Wissen über die Netzwerk-Struktur, die sich ein Angreifer erst mühevoll erarbeiten muss).

Die Art der Gegenmaßnahmen lassen sich in **Aktive** und **Passive** einteilen. Bei den zuerst Genannten wird versucht, einen direkten Kontakt zum Angreifer aufzubauen und zwar mit dem Ziel, einen **Gegenangriff** durchzuführen. Das **Opfer** wird so zum **Angreifer**, die damit verbundenen rechtlichen Probleme werden im Abschnitt ▶ rechtliche Aspekte behandelt. Normalerweise werden nur passive Aktionen ausgeführt, am Besten nur solche, die vom Angreifer unbemerkt bleiben. (Vorteil: der Angreifer setzt den Angriff fort und gibt damit möglicherweise neue wichtige Informationen z. B. über seine Identität preis.) Zu diesen Aktionen gehören beispielsweise:

* Anpassen der Firewall-Regeln (Paketfilter),
* Ändern der Prozessprioritäten (DOS-Gegenmaßnahme),
* Erhöhung der aufzuzeichnenden Daten,
* Dienste beenden, Ports sperren oder vom Netzwerk trennen (nur im Extremfall und wohlüberlegt).

Ein Neustart des Systems sollte wohlüberlegt erfolgen, da unter Umständen wichtige Informationen verloren gehen. Lesen Sie dazu unbedingt das Kapitel Grundlagen Sicherheit.

2.4 Aufbau eines IDS

Um die oben beschriebene Funktionalität zu implementieren, verwendet ein IDS in der Regel vier Komponenten, die je nach System in unterschiedlicher Ausprägung zu finden sind:

* **Ereigniskomponente:**
  Über Sensoren werden Daten aus der Umgebung aufgenommen, entsprechende Ereignisse generiert und nach einer Vorverarbeitung (preprocessing) an die Analysekomponente weitergeleitet. Die Sensoren können aus unterschiedlichen Quellen lesen (Logfiles, Betriebssystem-Ereignisse, Netzwerk-Verkehr, ...)

Entscheidend ist die Platzierung der Sensoren.

* **Sicherungs- bzw. Aufzeichnungskomponente:**
  Die Sicherungskomponente eines IDS ist für die Protokollierung der gewonnenen Erkenntnisse zuständig. Sie bildet damit die Grundlage für eine spätere Auswertung und die juristische Verwertbarkeit der Informationen.
* **Analysekomponente:**
  Die von der Ereigniskomponente gesammelten Daten werden analysiert. Beim Feststellen eines Angriffs wird Alarm ausgelöst, d. h. sowohl die Aufzeichnungskomponente als auch die Komponente für (halb-) automatische Gegenmaßnahmen werden in Kenntnis gesetzt. Für die Datenanalyse haben sich in den letzten Jahren die bereits oben erläuterten zwei Methoden "Missbrauchserkennung" und "Anomalieerkennung" herausgebildet.
* **Monitor und Aktionskomponente:**
  Die Ergebnisse der Analyse werden derart aufbereitet, dass der Administrator damit auch etwas anfangen kann. Unter Umständen werden Gegenmaßnahmen eingeleitet, automatisch, halbautomatisch oder manuell. Näheres dazu in "Gegenmaßnahmen und rechtliche Aspekte" und "Datenvisualisierung und eventuelle Einleitung von Gegenmaßnahmen".

3 Praktischer Einsatz und Effizienz von ID Systemen

In diesem Abschnitt werden praktische Aspekte behandelt. Dazu zählen prinzipielle Probleme, die den Einsatz von IDS behindern und nur teilweise oder gar nicht in den Griff zu bekommen sind. Nach einigen Betrachtungen zu Effektivität von ID-Systemen wird deren Eingliederung in ein Netzwerk diskutiert. Im Anschluss daran werden Angriffsmöglichkeiten auf IDS erörtert, Gegenmaßnahmen für Angriffe und deren rechtliche Konsequenzen vorgestellt und weiterführende Hinweise gegeben.

3.1 Prinzipielle Probleme

Ein grundsätzlich unlösbares Problem stellt der Einsatz von Verschlüsselungstechnologien dar. Verschlüsselung soll unberechtigten Personen den Zugang zu Informationen verwehren. Berechtigt sind alle diejenigen, die ein Geheimnis kennen und dies auch beweisen können (Besitz eines Schlüssels). Ist das ID-System nicht berechtigt, die Daten zu lesen, kann es auch keine Angriffe mehr erkennen, der Zweck des IDS ist damit infrage gestellt. Bespiel: Bei verschlüsselten Dateisystemen sind dem IDS keinerlei Informationen über Dateigrößen, Dateirechte oder gar Dateiinhalte zugänglich. Es wird lediglich erkannt, ob Änderungen stattfanden, die betroffenen Dateien lassen sich jedoch nicht identifizieren. Nun könnte ein Ausweg darin bestehen, dem IDS den Schlüssel zur Verfügung zu stellen. Dies stellt jedoch eine massive Bedrohung für die zu schützenden Daten dar, da ein Angreifer nun das IDS als Zugangspunkt missbrauchen könnte.

Ein weiteres Hauptproblem für NIDS liegt in der Skalierbarkeit, oder anders ausgedrückt im exponentiellen Wachstum der Durchsatzrate von Netzwerken. Eine Analyse in Echtzeit ist in Gigabit-Netzwerken mit entsprechend großem Verkehrsaufkommen nicht bzw. nur eingeschränkt möglich und zudem äußerst teuer. Eine eingeschränkte Analyse hat den entscheidenden Nachteil, dass u. U. nicht alle Angriffe als solche identifiziert werden und damit Sicherheit lediglich suggeriert wird.

3.2 Effektivität

Heutzutage werden oftmals missbrauchserkennende Systeme eingesetzt, weil diese einfacher zu realisieren sind. Der Hauptnachteil dieser Systeme besteht darin, dass sie lediglich bekannte Angriffe aufgrund einer Mustererkennung identifizieren können. Die Effektivität ist also stark abhängig von der Vollständigkeit der Signaturdatenbank sowie deren Aktualisierungsintervallen. Um so länger die Verbreitung neuer Angriffsmuster dauert, desto weniger Schutz kann ein missbrauchbasiertes IDS bieten.

Anomaliebasierte Systeme sind in realen Umgebungen aufgrund ihrer sehr hohen falsch positiven Fehlerrate kaum einsetzbar. Eine manuelle Bewertung, ob ein Angriff tatsächlich stattfand, ist oftmals undenkbar. Die Rate liegt derzeit selbst bei größter Anstrengung nur knapp unter 1%.

Die Effektivität ist weiterhin von der Qualität und Aussagekraft der gesammelten Auditdaten abhängig. Meist ist die Menge der auszuwertenden Daten jedoch so groß, dass es kaum möglich ist, eine intensive Analyse in Echtzeit durchzuführen. Ansätze zur Verteilung dieser Lasten auf mehrere Knoten sind Bestandteil heutiger Forschungsarbeiten der Technischen Universität Cottbus, die mit der HEIDI-Architektur sogar schon einen Prototypen aufzuweisen hat.

Die Integration host- und netzwerkbasierter IDS ist ein weiteres aktuelles Ziel, welchem sich Systeme wie prelude widmen.

Im Abschnitt "Charakteristika eines guten IDS" wird jedoch deutlich, dass die Anforderungen an Intrusion Detection Systeme nur schwer in Einklang zu bringen sind. Deshalb genügen real existierende Systeme keinem universellen Anspruch. Aussagen über die Effektivität sind sehr kritisch zu bewerten.

3.3 Eingliederung in die lokale Netzwerkstruktur

Intrusion Detection Systeme werden oftmals mit einer Firewall in Verbindung gebracht, manchmal sogar mit dieser verwechselt. Ein IDS kann jedoch eine Firewall nicht ersetzen, sondern stellt lediglich eine sinnvolle Ergänzung zu ihr dar.

Nun stellt sich jedoch die Frage, an welchen Stellen Firewall und IDS günstigerweise im Netzwerk platziert werden. Wird es vor der Firewall positioniert, kann es den hereinkommenden Netzwerkverkehr auf Angriffe prüfen. Jedoch stellt nicht nur der Verkehr, der von Außen in ein Netzwerk dringt, eine Gefahr dar, sondern auch der Verkehr vom Inneren eines Netzwerkes. Angriffe gehen laut CERT Studie zu mehr als 50 Prozent vom Netzwerkinneren aus, oftmals von eigenen Mitarbeitern oder von externen Dienstleistern mit innerbetrieblichem Netzzugang. Dabei kann ein Angriff auch ungewollt durch Viren und Würmer durchgeführt werden. Deshalb ist eine Sicherheitspolitik unumgänglich. In der dort beschriebenen Sicherheitsarchitektur besitzt ein Firewallsystem beispielsweise die Aufgabe, den Verkehr zwischen Netzwerksegmenten zu steuern. Es wird also festgelegt, welcher Verkehr aus einem externen Netz, sei es das Internet oder ein Netzwerk mit anderem Sicherheitsbedarf, in den zu schützenden Netzwerkbereich gelangen darf. Ein IDS hingegen ist keine Steuerungskomponente, sie hat vielmehr eine Kontrollfunktion. Es soll Angriffe auf das System feststellen und somit die Durchsetzung der Sicherheitspolitik gewährleisten.

Die Platzierung eines IDS im zu schützenden Netzwerksegment erscheint genauso sinnvoll wie vor dem Netzwerkbereich - eine Entscheidung hängt sehr stark vom Sicherheitsbedarf, von den durch die Sicherheitspolitik festgelegten Restriktionen und von der Sicherheitsarchitektur insgesamt ab. Besonders schützenswerte Bereiche wie Forschung und Entwicklung sollten mit zwei ID-Systemen überwacht werden.

Für die Sensoren eines Netzwerk-IDS gibt es ein prinzipielles Problem. Heutzutage werden fast ausschließlich **Switches** eingesetzt - das heißt, die Kommunikation zwischen den Netzwerkpartnern wird geschaltet und erfolgt über Punkt-zu-Punkt Verbindungen (wie beim Telefon). Ist das IDS an einem normalen Port des Switches angeschlossen, kann es somit nicht den gesamten Netzwerkverkehr abhören, sondern es erhält lediglich die Pakete, die an ihn adressiert sind. Viele Switches (vor allem im höheren Preissegment ab 500 Euro) bieten allerdings Monitoring-Ports, auf die der gesamte Netzwerkverkehr, der den Switch passiert, geleitet wird. Es versteht sich von selbst, dass ein NIDS-Host an einem solchen Port betrieben werden muss. Die damit verbundenen Durchsatz-/Performance Probleme sind im praktischen Einsatz jedoch zu beachten (siehe Prinzipielle Probleme).

3.4 Das IDS als Angriffspunkt

Sicherheitskomponenten wie Firewall-Systeme oder IDS sind besonders sicherheitskritisch. Meist versucht ein Angreifer diese zu manipulieren oder unbrauchbar zu machen. Deshalb müssen solche Systeme geschützt werden bzw. selbst resistent gegenüber etwaigen Angriffen sein.

Besonders zu schützen sind die Aufzeichnungsdaten. Um Manipulationen bereits bestehender Einträge zu unterbinden, ist sicherzustellen, dass die Datei nur wachsen kann (s. "Erweiterte ext2-Dateiattribute", samhain, bzw. tripwire). Des Weiteren ist es denkbar, dass ein Angreifer bewusst massenhaft Aktionen ausführt, deren Protokollierung tausende wenn nicht millionenfache Einträge nach sich ziehen. Das Herausfinden eines Angriffes gleicht dann dem Suchen einer Nadel in einem Heuhaufen. Im Extremfall laufen die Logfiles über und der Aufzeichnungsdienst steht nicht mehr zur Verfügung (DOS-Angriff).

Bei signaturbasierten NIDS sind vor Allem Angriffe festzustellen, welche Techniken wie **Insertion** und **Evasion** verwenden. Ein Angreifer fügt in den vom IDS überwachten Datenstrom Pakete ein, und versucht durch diese Verwirrungstaktik den Angriff zu verschleiern.

3.4.1 Insertion - Einfügen ungültiger Daten

Insertion basiert darauf, dass ein IDS Pakete akzeptiert, die das **normale Endsystem** normalerweise nicht annehmen würde. Das Intrusion Detection System geht also fälschlicherweise davon aus, dass ein Paket vom entsprechenden Host akzeptiert wird. Beim Insertion-Angriff wird diese Tatsache ausgenutzt und gezielt solche Pakete versendet, die ausschließlich vom IDS akzeptiert werden. Dadurch lassen sich beispielsweise signaturbasierte Systeme folgendermaßen austricksen: Angenommen, das IDS reagiert auf die Sequenz **ATTACK**. Der Angreifer versendet nun die Sequenz **AXTTAXXCXXK**, jedoch sind die **X** Pakete derart manipuliert, dass sie nur vom IDS akzeptiert werden. Die Einbruchserkennung stellt keinen Angriff fest, weil ja das Muster **ATTACK** nicht passt. Das Zielsystem jedoch verwirft die **X**-Pakete, ein Angriff findet dort tatsächlich statt.

3.4.2 Evasion - Ablehnen gültiger Daten

Die zweite Möglichkeit - **Evasion** - arbeitet komplementär zur soeben vorgestellten **Insertion**-Technik. Hier lehnt das IDS Pakete ab, die das Endsystem durchaus akzeptiert. Die Systemschwachstelle ist also bei Insertion und Evasion identisch: Das IDS **sieht** einen anderen Datenstrom als das Zielsystem. Sind also bspw. bei der Sequenz **ATTACK** die **T**-Zeichen manipuliert, so würde das IDS lediglich **AACK** erkennen und dementsprechend kein passendes Angriffsmuster zuordnen können.

3.5 Gegenmaßnahmen und rechtliche Aspekte

Wie bereits im Abschnitt "Datenvisualisierung und eventuelle Einleitung von Gegenmaßnahmen". erwähnt, lassen sich die Gegenmaßnahmen in Aktive und Passive unterteilen.

Nutzt man die Daten, um aktiv einen Gegenangriff einzuleiten, stellt sich die Frage, was man letzten Endes damit bewirkt und ob dies wirklich den Sinn eines Intrusion Detection Systems entspricht. Durch aktive Maßnahmen erfährt der Eindringling, dass er entdeckt wurde und wird sich in der Regel zurückziehen, um dann zu einem späteren Zeitpunkt mit ausgefeilteren Mitteln erneut anzugreifen. Das Sammeln weiterer Daten über den Angriff bzw. die Identität des Angreifers wird damit also verhindert und der Angreifer wird gleichzeitig gewarnt. Deshalb sollte der Einsatz aktiver Maßnahmen nur in Ausnahmefällen erfolgen. Im Idealfall zeigt das IDS also überhaupt keine Reaktion nach Außen, sondern geht seiner eigentlichen Aufgabe nach, nämlich der Sammlung von Informationen über Angriff und Angreifer sowie Benachrichtigung des Administrators (siehe Abschnitt "Datenvisualisierung und eventuelle Einleitung von Gegenmaßnahmen").

Bei Gegenmaßnahmen ist stets darauf zu achten, dass der Angreifer die Auditdaten bewusst manipulieren und

seine Identität fälschen kann (Maskerade). Er verwendet dazu beispielsweise Spoofing-Techniken zum Fälschen der IP-Adresse. Dies festzustellen, ist äußerst schwer. Weiß der Angreifer von den Gegenmaßnahmen, so kann er anderen Personen dadurch schaden, dass er unter deren Identität einen Angriff durchführt.

Die gesammelten Auditdaten stellen nach §416 der Zivilprozessordnung kein rechtsverbindliches Beweismittel dar, wie z. B. ein notariell beglaubigtes Schriftstück. Die Daten unterliegen der freien Beweisführung, womit sie den gleichen gerichtlichen Status wie eine Zeugenaussage haben und ihre Beurteilung im Ermessen des Gerichtes liegt. Der Grund für die Einordnung als **nicht rechtsverbindliches Beweismittels** gründet sich auf die Tatsache, dass die Auditdaten nachträglich modifiziert werden können und somit als nicht manipulationssicher gelten. Um den Beweiswert zu erhöhen, sollten die Auditdaten vertrauenswürdig sein, es bedarf also der Sicherstellung der Integrität (z. B. durch eine digitalen Signierung der gesammelten Daten durch das IDS). Die dazu verwendeten Maßnahmen sollten aber ebenfalls vor Gericht nachgewiesen werden können (z. B. die Geheimhaltung des privaten Schlüssels, Vertrauenswürdigkeit).

Es soll noch erwähnt werden, dass personenbezogene Daten nach den Datenschutzgesetzen nur in annonymisierter Form gespeichert werden dürfen.

3.6 Sonstige Hinweise

Wie bereits erwähnt, ist das Gebiet der Anomalieerkennung noch sehr jung, es stehen kaum praktische Erfahrungen zur Verfügung. Besonders problematisch ist die Erstellung des **normalen** Systemprofils, bei NIDS und heterogenen Netzwerken ist dies aufgrund der großen Protokollvielfalt äußerst schwierig - meistens sogar unmöglich. Bei sich selbst anpassenden Systemen kann der Angreifer die Überwachung derart beeinflussen, dass potentielle Angriffe in das Systemprofil übernommen werden. Außerdem interpretieren restriktiv eingestellte Systeme normalen Netzwerkverkehr fälschlicherweise als Angriff, sodass die Gefahr besteht, dass der Administrator Angriffsmeldungen einfach ignoriert. Es werden deshalb große Anstrengungen in der Forschung unternommen, um diese falsch-positiven Ergebnisse zu vermeiden. Zusammenfassend kann man sagen, dass das **Lernen** des Normalzustandes die Qualität der Ergebnisse maßgeblich bestimmt.

Missbrauchserkennungssysteme wie `snort` und samhain sind entgegen ihren anomaliebasierten Pendants schon häufig und erfolgreich im Einsatz. Bei diesen Systemen ist die Aktualität der Signaturen von wichtiger Bedeutung. Man muss sich bewusst sein, dass nur bereits bekannte Angriffe erkannt werden.

Ein IDS darf natürlich so wenig wie möglich Angriffsfläche bieten (s. Abschnitt "Das IDS als Angriffspunkt"). Oft stellen Sicherheitskomponenten wie Firewall oder IDS das erste Ziel eines Angriffs dar. Sind diese erst einmal ausgeschaltet, kann ein Cracker ungestört (da meist lange unbemerkt) Schaden anrichten. Darum heißt es bei einem IDS wie bei allen Sicherheitskomponenten: **So viele Dienste wie nötig, aber so wenige wie möglich!** Je weniger Ports offen sind, je weniger Dienste angeboten werden und je weniger Benutzerkonten (s. "Über die Sicherheit von Passwörtern") existieren, desto resistenter ist ein Host gegenüber Angriffen.

Es ist wichtig, dass die Integrität der Konfigurationsdaten gesichert ist. Da einige Dateien sicherheitsrelevante Informationen enthalten, sollten diese vertraulich behandelt werden (auch kein Lesezugang für Benutzergruppe). Signaturdateien sind auch vor Veränderung zu schützen und sollten am Besten vertraulich behandelt werden

Besondere Vorsicht ist auch beim Protokollieren geboten, da ein Überlaufen der Protokolldateien von einem Angreifer gezielt ausgenutzt werden könnte (lässt erst die Protokolldateien überlaufen und führt erst dann den eigentlichen Angriff aus).

4 Charakteristika eines guten IDS

Folgende Aspekte stellen eine qualitative Bewertungsgrundlage für ID-Systeme dar:

* Erkennung von Abweichungen bezüglich des normalen Systemverhaltens: Dabei sollten möglichst keine falsch positiv/negativen Ergebnisse geliefert werden. Das setzt voraus, dass das IDS resistent gegenüber Täuschungsversuchen ist.
* Kontinuität und Resistenz: Das System läuft **verlässlich** ohne Eingriffe des Administrators. Das bedeutet auch, dass das IDS selbst resistent gegenüber Angriffen ist.
* Fehlertoleranz und Flexibilität: Da Einsatzweck und/oder Umgebungsparameter durchaus variabel sind, soll sich das IDS an die Systemumgebung anpassen können. Beispielsweise darf die Installation neuer Software zu keinen ungewünschten Effekten führen. Das IDS sollte auch mit irrelevanten Fehlersituationen umgehen können.
* Ressourcenverbrauch und Skalierbarkeit: Das IDS sollte zukunftssicher sein und sich an steigende Anforderungen bei laufendem Betrieb anpassen können.

Weitere Anforderungen wurden in der BSI-Studie "Intrusion Detection Systeme" aufgezeigt. Die Realisierung dieser Anforderungen ist nahezu undenkbar, real existierende IDS erfüllen nur wenige dieser Forderungen.