

Philipp Winter, Harald Lampesberger, Markus Zeilinger, Eckehard Hermann

Anomalieerkennung in Computernetzen

Seit Dekaden wird bereits an Anomalieerkennung in Computernetzen geforscht. Maßgebliche Erfolge blieben bis heute allerdings aus. Zwar werden regelmäßig Verfahren publiziert, die auf dem Papier viel versprechende Ergebnisse bringen, doch kaum eines schafft es, auch in der Praxis Einsatz zu finden. Der Beitrag zeigt die Gründe dafür auf und stellt vor, wie diesem Phänomen begegnet werden kann.



Philipp Winter, MSc

Wissenschaftlicher Mitarbeiter an der FH-Oberösterreich, Österreich

E-Mail: philipp.winter@fh-hagenberg.at



Harald Lampesberger, MSc

Wissenschaftlicher Mitarbeiter an der FH-Oberösterreich, Österreich

E-Mail: harald.lampesberger@fh-hagenberg.at



Dr. Markus Zeilinger

Fakultät für Informatik, Kommunikation und Medien, FH-

Oberösterreich, Österreich

E-Mail: m.zeilinger@fh-hagenberg.at



Prof. Eckehard Hermann

Fakultät für Informatik, Kommunikation und Medien, FH-

Oberösterreich, Österreich

E-Mail: eckehard.hermann@fh-hagenberg.at

1 Einleitung

Zur Erkennung von Angriffen in Computernetzwerken (*Intrusion Detection*), existieren zwei grundsätzliche Ansätze: die signaturbasierte Erkennung (*Misuse* oder *Signature Detection*) und die Anomalieerkennung (*Anomaly Detection*).

Signaturbasierte Systeme suchen mit Hilfe von Signaturen nach exakt definierten und bekannten Angriffsmustern. Signaturen sind hier spezielle Byte-Sequenzen, die Teil eines Angriffs sind. Die Falschalarmrate solcher Systeme ist, ausreichend präzise Signaturen vorausgesetzt, sehr gering. Etablierte signaturbasierte Systeme sind Snort [1] und Bro [2].

Jedoch hat das Konzept der signaturbasierten Erkennung mit einigen Schwierigkeiten zu kämpfen. Bisher unbekannte Angriffe können auf diesem Weg nicht erkannt werden, da hierfür keine Signaturen existieren. Signaturen müssen von Experten laufend gepflegt werden, um auf neue Angriffsmöglichkeiten zu reagieren und damit gute Erkennungsraten zu gewährleisten.

Anomalieerkennung versucht im Gegensatz zur signaturbasierten Erkennung nicht spezifische Angriffe, sondern Abweichungen vom Normalen (Anomalien) zu erkennen. Dies erscheint auf den ersten Blick reizvoll, da sich folgende Vorteile gegenüber signaturbasierter Erkennung ergeben:

- ◆ Neben bekannten können auch bisher unbekannte Angriffe erkannt werden, so sich diese signifikant von normalem Datenverkehr unterscheiden.
- ◆ Es ist kein exaktes Wissen über die Struktur und Eigenschaften von Angriffen erforderlich.
- ◆ Es ist nicht notwendig, laufend Signaturen für neue Angriffe einzuspeisen.

Im Sinne der Früherkennung von neuen Bedrohungen ist Anomalieerkennung auch als zentrale Sensorik in der IT-Frühwarnung zu sehen.

Die Anomalieerkennung in der *Intrusion Detection* mag zunächst als sehr junges Forschungsgebiet erscheinen. Allerdings wurde das Konzept bereits im Jahre 1986 von Dorothy Denning [3] beschrieben. Das Ergebnis ihrer Forschung wurde im Rahmen der Softwareprojekte IDES und im Nachfolger NIDES implementiert.

Während Denning noch mit User-Profilen und Statistik arbeitete, wurden im Laufe der Jahre zahlreiche weitere Konzepte vorgeschlagen und umgesetzt. Dazu gehören unter anderem Entscheidungsbäume, Markov-Modelle, neuronale Netze, Verfahren aus der Informationstheorie und maschinelles Lernen. Doch keiner der vielen Ansätze konnte sich bisher in kommerziellen Produkten und damit in der Praxis durchsetzen.

In diesem Beitrag wird diskutiert, was die gravierendsten derzeitigen Probleme in der Anomalieerkennung sind. Des Weiteren werden mögliche Auswege aus der Misere angeboten.

2 Probleme in der Anomalieerkennung

Im Folgenden werden Ursachen für die chronische Erfolglosigkeit von Anomalieerkennung in der Praxis erläutert (nach [4]).

2.1 Falschalarme verursachen hohe Kosten

Ein wesentliches Problem sind die Kosten, die durch hohe Falschalarmraten

(FAR) entstehen. Von einem Falschalarm wird gesprochen, wenn ein Intrusion Detection System (IDS) Datenverkehr fälschlicherweise als anormal (Angriff) klassifiziert. In einem solchen Fall muss der Alarm von einem Analysten (Arbeitszeit, Kosten) bearbeitet werden, um festzustellen, dass es sich eigentlich um Normalverkehr handelt.

Des Weiteren finden in der Regel viel weniger Angriffe als normale Ereignisse statt. Dadurch können selbst bei einer sehr niedrigen FAR die Falschalarme numerisch dominieren, und in Folge dessen wird die effektive Aussagekraft des IDS gesenkt.

Axelsson [5] beschreibt dieses Problem und hält eine Rate von maximal einem Fehlalarm in 100.000 Ereignissen als wesentlich für den Erfolg eines IDS.

2.2 Normal ist nicht gleich normal

Zentral ist auch die Frage, was überhaupt „normal“ ist. Das Normale definiert sich aus dem Kontext der Umgebung (z. B. Unternehmensnetzwerk, Anwendungsbereich, ...). Häufig wird dieser Umstand in der Forschung ignoriert und davon ausgegangen, dass Normalität in verschiedenen Netzwerken ähnlich aussieht. In der Praxis kann aber das, was für ein Netzwerk normal ist, in einem anderen eine missbräuchliche Verwendung oder ein Angriff sein.

Als Beispiel kann die Nutzung von P2P-Anwendungen genannt werden. Untersagt ein Unternehmen die Nutzung von P2P-Anwendungen nicht generell, aber das System zur Anomalieerkennung meldet ein anormales Ereignis bei Nutzung solcher Anwendungen, dann ist dies wenig sinnvoll und in diesem Kontext als Falschalarm zu sehen.

2.3 Auf die Daten kommt es an

Die Möglichkeiten zur Erkennung von Angriffen und Anomalien hängen wesentlich von den zur Verfügung stehenden Daten und deren Attributen ab. Nicht alle Arten von Angriffen sind in jeder Art von Netzwerkdaten zu erkennen. Arbeitet ein System zur Anomalieerkennung beispielsweise mit NetFlow-Daten, ist es allgemein damit nicht möglich, Anomalien in HTTP Nutzdaten zu erkennen, da in NetFlow Daten keine Nutzdaten enthalten sind.

2.4 Was bedeutet eine Anomalie eigentlich?

Systeme zur Anomalieerkennung beschränken sich häufig darauf, Anomalien als Abweichungen vom Normaldatenverkehr zu erkennen und an den Analysten zu melden. Dabei ist es zunächst unerheblich, ob es sich bei der Anomalie tatsächlich um einen Angriff oder lediglich um ungewöhnliche, aber harmlose Netzwerkaktivität (Änderung im Routing, Einführung eines neuen Netzwerkdienstes) handelt. Netzwerk-Administratoren müssen dann erst durch aufwändige Analysen des Netzwerkverkehrs herausfinden, was die Anomalie auslöste.

Im Kontext von Intrusion Detection ist das aber zu wenig, da Intrusion Detection der Erkennung von Angriffen dienen soll. Häufig bietet das System zur Anomalieerkennung dem Analysten auch nicht die notwendigen Informationen, um ein Ereignis korrekt einschätzen zu können.

2.5 Evaluierung

Eine umfassende Evaluierung eines neuen Systems zur Anomalieerkennung ist unerlässlich, um dessen Praxistauglichkeit festzustellen.

Hierfür sind Trainings- bzw. Testdaten notwendig, die möglichst nahe an der Realität eines Netzwerks sind und sowohl Normales als auch Anormales (Angriffe) enthalten. Dabei muss für alle Elemente der Datensätze klar annotiert sein, ob es sich um Normalverhalten oder eine Anomalie (Angriff) handelt.

Es sind zwei Varianten zur Erzeugung von Datensätzen denkbar, wobei beide die dargestellten Anforderungen kaum erfüllen können:

- ♦ Um der Forderung nach Realitätsnähe gerecht zu werden, können Daten aus einem Produktivnetzwerk (z. B. Provider-Netzwerk) als Ausgangsbasis verwendet werden. Diese Daten müssen mit Expertenwissen sorgfältig analysiert und annotiert werden.

Aufgrund der Komplexität von Netzwerkdaten ist dies aber ein schwieriger und zeitaufwändiger Prozess. Als weiteres Hemmnis erweisen sich hierbei Datenschutzaspekte, da nur schwer sicherzustellen ist, dass die realen Netzwerkdaten keine schutzwürdigen Informationen enthalten. Aus diesem Grund werden derartige Datensätze häufig nicht veröffentlicht.

- ♦ Als Alternative können Daten in künstlich gebauten Netzwerken mittels Simulation synthetisch erzeugt werden. Damit ist vollständig kontrollierbar, welcher Datenverkehr normal ist und welcher auf Anomalien und Angriffe hinweist.

Die Schwäche dieses Ansatzes liegt in der Komplexität moderner Netzwerke, welche einen realitätsnahen Nachbau sehr erschwert. Testergebnisse auf Basis solcher synthetischer Daten können daher nur bedingt auf ein reales Einsatzszenario übertragen werden.

Der bekannteste synthetisch erzeugte Datensatz ist der DARPA Datensatz von 1998 [6] und 1999 [7]. Hier wurden von den Lincoln Labs des MIT zur Evaluierung von Intrusion Detection Systemen Normalverhalten und Angriffe im Netzwerk einer fiktiven *Air Force Base* simuliert und aufgezeichnet. Der Datensatz gilt als einzigartig und wurde in vielen Forschungsprojekten verwendet und referenziert [8, 9, 10].

- ♦ Nichtsdestotrotz zeigt der Datensatz viele Unzulänglichkeiten [11, 12], die vor allem mit dem Umstand der Simulation zu tun haben. Zudem sind die Daten inzwischen über zehn Jahre alt und spiegeln weder im Bereich des Normalverhaltens noch bei den enthaltenen Angriffen die Realität in heutigen Netzwerken wieder.

3 Wie macht man es richtig?

Der vorhergehende Abschnitt lieferte einen kompakten Einblick in die zahlreichen und gravierenden Hürden, denen Forscher im Bereich der Anomalieerkennung gegenüberstehen. Der Zweck dieses Abschnitts ist es nun, Lösungsansätze für die soeben genannten Probleme anzubieten. Basierend auf [4] und [13] werden Empfehlungen gegeben, die trotz der fundamentalen Schwierigkeiten praxistaugliche Anomalieerkennung ermöglichen sollen. Diese Empfehlungen sind nicht als Checkliste gedacht, die nach Abarbeitung eine Erfolgsgarantie darstellt. Vielmehr sind sie als Anregungen und Tipps zu verstehen, die den richtigen Weg weisen sollen.

3.1 Problembereich eingrenzen

Die erste und auch wichtigste Empfehlung ist, den Problembereich, also die zu erken-

nenden Angriffe, so weit wie möglich einzugrenzen. In vielen Publikationen ist der Problembereich sehr vage und universell als „Anomalieerkennung“ definiert. Es erfolgt keine Festlegung auf spezielle Angriffsarten oder Protokolle. Dies ist ein häufiger Grund für das Scheitern der Bemühungen. Je genauer die zu erkennenden Angriffe eingegrenzt werden können (z. B. Spam-Aktivität oder P2P-basierte Botnetze), desto besser kann das zugrunde liegende Phänomen analysiert, verstanden und schließlich gelöst werden. Allumfassende Wunderlösungen sind in der Anomalieerkennung nicht realisierbar.

3.2 Falschalarmrate

Wie bereits erwähnt, ist ein weiterer Grund für das Scheitern von Forschungsvorhaben eine inakzeptabel hohe Falschalarmrate. Für die Entwickler von Systemen zur Anomalieerkennung gilt, dass die FAR eines Systems bei praktisch 0% liegen muss.

Aus diesem Grund sollten Systeme zur Anomalieerkennung immer in Hinsicht auf ihre FAR optimiert werden, obwohl dies die Erkennungsrate negativ beeinflussen kann. Während ein System mit einer verminderten Erkennungsrate immer noch von gewissem Nutzen ist, ist ein System mit einer hohen FAR für den Realbetrieb gänzlich untauglich.

Letztlich kann eine optimale Abstimmung von Erkennungsrate und FAR nur mithilfe einer Risikoanalyse und anschließender Akzeptanz von Risiken durch eine verminderte Erkennungsrate erfolgen.

3.3 Adäquate Datenquellen

Eine geeignete Datenquelle bietet das Fundament, auf dem ein System zur Anomalieerkennung aufbaut. Hierfür ist der betrachtete Problembereich entscheidend. Wenn klar definiert ist, welche Arten von Angriffen und Anomalien erkannt werden sollen, kann entschieden werden, welche Daten dafür notwendig sind. Ist es das Ziel, große *Distributed Denial of Service* (DDoS) Angriffe zu erkennen, können Flussdaten (NetFlow) hilfreich sein. Webbasierte Angriffe wie *SQL Injections* oder *Cross Site Scripting* können hingegen ausschließlich in Nutzdaten detektiert werden. Die Entwicklung neuer Methoden zur Anomalieerkennung kann daher schon bei der Wahl der Datenquelle scheitern.

3.4 Informationsbereitstellung

Um die Arbeit für Netzwerk-Administratoren zu erleichtern, muss ein System zur Anomalieerkennung im Kontext einer Anomalie so viele Informationen wie möglich zur Verfügung stellen. Es gilt, den Interpretationsspielraum möglichst einzugrenzen. Dadurch sinkt der Bearbeitungsaufwand und Netzwerk-Administratoren können ihre Zeit effizient nutzen, um auf die Ursachen der Anomalien zu reagieren. Des Weiteren können diese Informationen als Basis für Muster von signaturbasierten Erkennungssystemen dienen.

3.5 Datensätze

Wie bereits erwähnt, ist ein weiteres fundamentales Problem die Evaluierung von Systemen zur Anomalieerkennung. Diese verlangt möglichst aktuelle, realistische und annotierte Datensätze. Leider ist dafür in absehbarer Zeit keine einfache Lösung in Sicht. Da keine adäquaten Datensätze öffentlich verfügbar sind, müssen diese bei Bedarf selbst erzeugt werden. Die eigentliche Erzeugung eines Datensatzes hängt vom jeweiligen Szenario ab. Sollte es aber nicht möglich sein, einen zufrieden stellenden Datensatz zu erzeugen, muss abgewogen werden, ob weiterführende Forschung überhaupt zielführend ist, da keine gründliche Evaluierung gewährleistet ist.

3.6 Was soll gelernt werden?

Zudem lernen Systeme zur Anomalieerkennung häufig die Charakteristika von Normalverhalten. Das heißt, ein Modell wird mit normalen, also gutartigen Netzwerkdaten trainiert und Netzwerkdaten, die von diesem Normalverhalten abweichen, werden dementsprechend als anomal klassifiziert. Da das Normalverhalten wie bereits erwähnt stark variiert, kann es sich anbieten, das Konzept umzudrehen und stattdessen das „Aussehen“ von böartigen Netzwerkdaten zu lernen.

Auf diese Weise kann das System zur Anomalieerkennung Variationen von gelernten Angriffen erkennen. Außerdem impliziert ein Alarm des Systems dann bereits, dass es sich um einen Angriff oder eine Angriffs-Variation handelt. Würde allerdings das Normalverhalten gelernt, ist zunächst nicht klar, ob der Anomalie auch ein Angriff zugrunde liegt. In dieser Domäne existiert jedoch noch weiterer

Forschungsbedarf, um die Eigenschaften und Vorteile beider Ansätze besser verstehen zu können.

3.7 Empfehlungen

Nachfolgend sind die Empfehlungen dieses Abschnitts in kompakter Form zusammengefasst:

- ♦ Die Angriffe, die erkannt werden sollen, müssen sehr genau abgegrenzt und definiert werden.
- ♦ Die Falschalarmrate muss bei praktisch 0% liegen bzw. einer Risikoanalyse unterliegen. Jeder falsche Alarm senkt die effektive Aussagekraft des Systems.
- ♦ Eine geeignete Datenquelle muss verwendet werden. Die Informationen dieser Datenquelle müssen ebenso fundiert aufbereitet werden, damit die Analyseverfahren auch in der Lage sind, die gewünschten Angriffe erkennen zu können.
- ♦ Möglichst viele Informationen müssen bereitgestellt werden, damit die Ursache einer Anomalie schnell identifiziert und darauf reagiert werden kann.
- ♦ Statt dem Normalverhalten können auch Angriffe erlernt werden, um Variationen gelernter Angriffe erkennen zu können.

4 Anomalieerkennung in der IT-Frühwarnung

Die Arbeit der Autoren wird im Rahmen des *Early Warning Research Labs* durchgeführt [14]. Der Schwerpunkt der Forschungsgruppe liegt im Bereich der IT-Frühwarnsysteme. Das Konzept der Frühwarnung kann als Schutzmaßnahme für kritische Informationsinfrastrukturen eingesetzt werden und Anomalieerkennung wird in diesem Kontext als Erfolg versprechende Komponente gesehen.

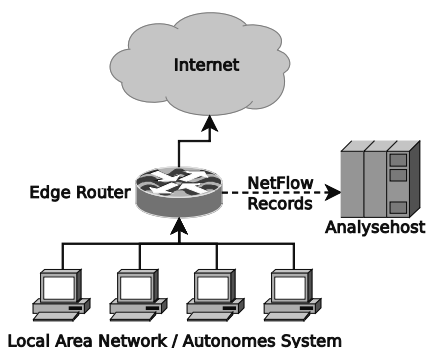
Ziel der aktuellen Forschungsaktivitäten ist ein Prototyp für praxistaugliche Anomalieerkennung in Hochgeschwindigkeits-Netzwerken. Dabei werden zwei unterschiedliche Ansätze verfolgt, die einander ergänzen. Der erste Ansatz analysiert flussbasierte Netzwerkdaten, während sich der zweite Ansatz mit der Analyse von Nutzdaten, mit Schwerpunkt HTTP, befasst. Beide Ansätze werden im Folgenden näher vorgestellt.

4.1 Anomalieerkennung in Flussdaten

Der erste Ansatz macht von flussbasierten NetFlow-Daten Gebrauch. NetFlow ist ein Netzwerk-Protokoll, das Metainformationen über Netzwerkverbindungen zur Verfügung stellt. Lang andauernde TCP-Verbindungen, die mitunter viele Gigabytes an Daten transportieren, können so zu NetFlow-Daten zusammengefasst werden, die lediglich einige wenige Bytes umfassen. Diese Leichtgewichtigkeit geht allerdings auf Kosten der verfügbaren Daten. Die Payload der ursprünglichen, rohen Netzwerkdaten geht verloren. Es stehen lediglich Metainformationen, wie beispielsweise Quell- und Ziel-IP-Adresse, Quell- und Ziel-Port und die in einer Verbindung transportierten Bytes und Pakete zur Verfügung. Aus diesem Grund ist zu beachten, dass damit nur Angriffe erkennbar sind, die sich auf der Netzwerkebene manifestieren (z. B. Spam-Aktivität, Scanning-Verhalten, Wurmasbrüche, DDoS-Angriffe und Botnetz-Aktivität).

Abbildung 1 verdeutlicht das Konzept. Ein Edge Router schickt NetFlow-Daten zur Analysekomponente, welche die NetFlow-Daten speichert und auswertet. Die Ergebnisse der Analyse werden in einer Weboberfläche visualisiert.

Abb. 1 | Flussdaten-Konzept



Ein Ausfall oder eine Fehlfunktion der Analysekomponente hat keinerlei Auswirkung auf das Produktivnetzwerk. Auch die Integration der Analysekomponente in ein bestehendes Netzwerk gestaltet sich als sehr einfach, da sie lediglich am Edge Router als NetFlow-Collector konfiguriert werden muss.

Die Analyse der Flussdaten erfolgt mit informationstheoretischen Verfahren, insbesondere der Shannon-Entropie. In Zeitserien wird mit der Entropie anomales Verhalten detektiert [15, 16]. Derartige Anomalien sind häufig Indikatoren

für Scan-Verhalten, Wurmasbrüche oder DDoS-Angriffe.

Zudem werden einfache Metriken zur Erkennung von Spam-Aktivität umgesetzt [17]. Diese sollen es ermöglichen, infizierte Hosts im internen Netzwerk zu detektieren, welche einem Botnetz zum Beispiel als Spam-Drohnen dienen.

4.2 Anomalieerkennung in Nutzdaten

Dieser Ansatz beschäftigt sich primär mit Anomalieerkennung in HTTP-Daten, da dieses Protokoll intensiv im Internet verwendet wird und Sicherheitslücken verstärkt in Webapplikationen auftreten [18]. Eine Eigenheit von HTTP ist, dass das Protokoll nur die groben Rahmenbedingungen des Informationstransfers vorschreibt, jedoch die eigentliche Applikationslogik in der Webapplikation stattfindet. Es ergeben sich unterschiedliche Angriffsvektoren auf Server und Applikation, und in der Folge kann bösartiger Schadcode in unterschiedlichen Protokollzuständen übertragen werden.

Anomalieerkennung in HTTP ist kein unergründetes Terrain. In den letzten Jahren gab es mehrere relevante Publikationen in diesem Forschungsfeld [19, 20, 21, 22, 23, 24]. Diese Veröffentlichungen stellen Konzepte vor, um Webapplikationen durch Anomalieerkennung zu schützen.

Bösartige Daten können vom Client primär in Form von GET/POST-Requests an die Applikation übertragen werden, deshalb beziehen sich die Analysen auf diese Methoden von HTTP. Ein Nachteil der vorhin referenzierten Konzepte ist, dass alle einen angriffsfreien Datensatz für das Training benötigen und damit mit dem Problem der Generierung von realitätsnahen Trainingsdaten zu kämpfen haben (siehe Probleme der Anomalieerkennung). Der praktische Einsatz ist aus diesem Grund eingeschränkt.

Der von den Autoren angestrebte Ansatz basiert auf der Annahme, dass bösartige Ereignisse wesentlich seltener stattfinden als normale Ereignisse. In erster Instanz wird die zu analysierende Datensequenz durch ein Markov-Modell mit variabler Ordnung bewertet [25]. Diese Methode stammt ursprünglich aus dem Bereich der verlustfreien Kompression und das Modell liefert in diesem Anwendungsfall die Auftrittswahrscheinlichkeit der betrachteten Datensequenz. So werden seltene Sequenzen mit einer niedrigen

und oft vorkommende Sequenzen mit einer hohen Wahrscheinlichkeit bewertet.

In zweiter Instanz werden Datensequenzen mit einer stark abweichenden Wahrscheinlichkeit als Anomalie klassifiziert. In den bisher durchgeführten Tests wiesen oft auftretende Datensequenzen ähnlich verteilte Wahrscheinlichkeiten auf. Durch robuste Ausreißererkennung in der Zeitserie der vergangenen Wahrscheinlichkeitswerte wurden in den Tests viel versprechende Ergebnisse erzielt.

Eine gefundene Anomalie bedeutet noch nicht, dass es sich um einen Angriff handelt. Deshalb ist in der dritten Instanz geplant, ähnliche Anomalien zu gruppieren. Einem Administrator ist es dadurch möglich, bestimmte Anomaliegruppen vom Reporting auszunehmen oder den Schweregrad von bestimmten Gruppen anzupassen. Die Falschalarmrate soll so zusätzlich gesenkt und die effektive Aussagekraft des Systems gesteigert werden können.

Die Herausforderungen in dem vorgestellten Konzept sind das automatische und korrekte Lernen des Markov-Modells und die Durchsatzleistung. Automatisches Lernen ist wichtig, da in realistischen Szenarien nicht von angriffsfreien Trainingsdaten ausgegangen werden kann. Zusätzlich muss für einen praktischen Einsatz die Möglichkeit bestehen, normale Datensequenzen oder Angriffssequenzen nachträglich zu lernen. Die Analyse von Nutzdaten ist rechenintensiv, da IP-Pakete defragmentiert, TCP-Verbindungszustände verarbeitet und Nutzdaten analysiert werden müssen. Um trotzdem eine akzeptable Durchsatzleistung zu gewährleisten, wird eine Parallelisierung im Softwareprototyp angestrebt.

Abb. 2 | Nutzdaten-Konzept

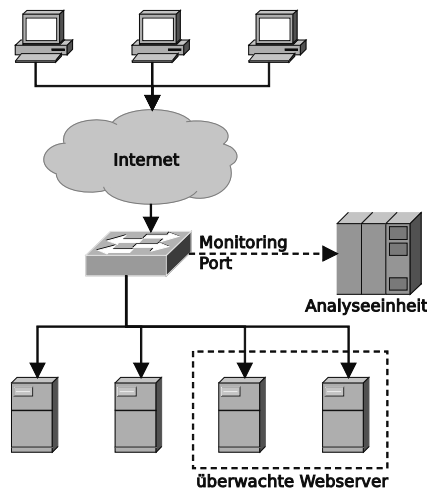


Abbildung 2 zeigt das Einsatzszenario des geplanten Softwareprototypen. Der Netzwerkverkehr, der für die zu überwachten Server bestimmt ist, wird passiv an die Analyseeinheit weitergeleitet, welche die Anomalieerkennung und Alarmierung durchführt.

5 Zusammenfassung

In diesem Beitrag wiesen wir zunächst auf die Probleme hin, mit denen nahezu alle Forschungsaktivitäten in diesem Bereich zu kämpfen haben. Diese Probleme sind so fundamentaler Natur, dass die meisten Forschungsergebnisse in der Anomalieerkennung in der Praxis nicht anwendbar sind.

Daraufhin wurden Empfehlungen und Wegweiser genannt, mit denen den bekannten Schwierigkeiten zumindest aus dem Weg gegangen werden kann. Schließlich wurde die Arbeit der Autoren im Kontext des *Early Warning Research Labs* kurz vorgestellt.

Abschließend kann behauptet werden, dass das Gebiet der Anomalieerkennung trotz gravierender Hürden und wenig Fortschritten in den letzten Jahren nach wie vor sehr attraktiv und viel versprechend ist. Allerdings muss die Forschungsgemeinschaft beginnen, etablierte Paradigmen zu überdenken und auch völlig neu zu definieren.

So ist es beispielsweise dringend notwendig, dass neu erarbeitete Verfahren und Konzepte ausführlich und fundiert getestet werden. Denn das eigentliche Ziel darf nicht aus den Augen verloren werden: Verfahren müssen im Produktivbetrieb einen tatsächlichen Mehrwert liefern. Erst wenn dies der Fall ist, kann Forschung tatsächlich als erfolgreich bezeichnet werden.

Solange dies nicht geschieht, dreht sich die Forschung in der Anomalieerkennung unweigerlich im Kreis und signifikante und praxistaugliche Ergebnisse werden weiterhin ausbleiben.

Literaturverzeichnis

[1] Snort, <http://www.snort.org>.

- [2] Bro Intrusion Detection System, <http://www.bro-ids.org>.
- [3] D. E. Denning, „An intrusion-detection model,” *IEEE Trans. Softw. Eng.*, vol. 13, pp. 222–232, February 1987.
- [4] R. Sommer and V. Paxson, „Outside the closed world: On using machine learning for network intrusion detection,” *IEEE Symposium on Security and Privacy*, pp. 305–316, 2010.
- [5] S. Axelsson, „The base-rate fallacy and its implications for the difficulty of intrusion detection,” in *CCS '99: Proceedings of the 6th ACM conference on Computer and Communications Security*. New York, NY, USA: ACM, 1999, pp. 1–7.
- [6] D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall, D. Mcclung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham, and M. A. Zissman, „Evaluating intrusion detection systems: The 1998 darpa off-line intrusion detection evaluation,” in *Proceedings of the 2000 DARPA Information Survivability Conference and Exposition*, 2000, pp. 12–26.
- [7] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, „The 1999 darpa off-line intrusion detection evaluation,” *Computer Networks*, vol. 34, no. 4, pp. 579–595, 2000.
- [8] M. V. Mahoney and P. K. Chan, „Learning non-stationary models of normal network traffic for detecting novel attacks,” in *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2002, pp. 376–385.
- [9] M. V. Mahoney, „Network traffic anomaly detection based on packet bytes,” in *SAC '03: Proceedings of the 2003 ACM symposium on Applied computing*. New York, NY, USA: ACM, 2003, pp. 346–350.
- [10] Y.-I. Zhang, Z.-g. Han, and J.-x. Ren, „A network anomaly detection method based on relative entropy theory,” in *ISECS '09: Proceedings of the 2009 Second International Symposium on Electronic Commerce and Security*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 231–235.
- [11] M. V. Mahoney and P. K. Chan, „An analysis of the 1999 darpa/lincoln laboratory evaluation data for network anomaly detection,” in *Proceedings of the Sixth International Symposium on Recent Advances in Intrusion Detection*. Springer, 2003, pp. 220–237.
- [12] J. McHugh, „Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory,” *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 4, pp. 262–294, 2000.
- [13] C. Gates and C. Taylor, „Challenging the anomaly detection paradigm: a provocative discussion,” in *Proceedings of the 2006 workshop on New security paradigms*, ser. NSPW '06. New York, NY, USA: ACM, 2007, pp. 21–29.
- [14] Early Warning Research Lab (ewrl), <http://www.fruehwarnung.at>.
- [15] A. Wagner and B. Plattner, „Entropy based worm and anomaly detection in fast ip networks,” in *Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 172–177.
- [16] G. Nychis, V. Sekar, D. G. Andersen, H. Kim, and H. Zhang, „An empirical evaluation of entropy-based traffic anomaly detection,” in *Proceedings of the 8th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '08. New York, NY, USA: ACM, 2008, pp. 151–156.
- [17] A. Sperotto, G. Vliet, R. Sadre, and A. Pras, „Detecting spam at the network level,” in *Proceedings of the 15th Open European Summer School and IFIP TC6.6 Workshop, EU-NICE 2009*, Barcelona, ser. Lecture Notes in Computer Science, vol. 5733. Berlin: Springer Verlag, August 2009, pp. 208–216.
- [18] 2010 CWE/SANS Top 25 Most Dangerous Software Errors, <http://cwe.mitre.org/top25/>.
- [19] C. Kruegel and G. Vigna, „Anomaly detection of web-based attacks,” in *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2003, pp. 251–261.
- [20] K. Wang and S. J. Stolfo, „Anomalous payload-based network intrusion detection,” in *Recent Advances in Intrusion Detection*, ser. Lecture Notes in Computer Science, vol. 3224. Springer Berlin / Heidelberg, 2004, pp. 203–222.
- [21] K. Wang, J. J. Parekh, and S. J. Stolfo, „Anagram: A content anomaly detector resistant to mimicry attack,” in *Recent Advances in Intrusion Detection*, ser. Lecture Notes in Computer Science, vol. 4219. Springer Berlin / Heidelberg, 2006, pp. 226–248.
- [22] R. Perdisci, D. Ariu, P. Fogla, G. Giacinto, and W. Lee, „Mcpad: A multiple classifier system for accurate payload-based anomaly detection,” *Computer Networks*, vol. 53, no. 6, pp. 864–881, 2009, traffic Classification and Its Applications to Modern Networks.
- [23] Y. Song, A. D. Keromytis, and S. J. Stolfo, „Spectrogram: A mixture-of-markov-chains model for anomaly detection in web traffic,” in *Proc. of Network and Distributed System Security Symposium (NDSS)*, 2009.
- [24] T. Krueger, C. Gehl, K. Rieck, and P. Laskov, „Tokdoc: a self-healing web application firewall,” in *SAC '10: Proceedings of the 2010 ACM Symposium on Applied Computing*. New York, NY, USA: ACM, 2010, pp. 1846–1853.
- [25] R. Begleiter, R. El-Yaniv, and G. Yona, „On prediction using variable order markov models,” *J. Artif. Int. Res.*, vol. 22, no. 1, pp. 385–421, 2004.