Computernetzwerke gegen unbekannte Angriffe schützen

Autor(en): **Tellenbach, Bernhard / Brauckhoff, Daniela / Plattner, Bernhard**

Objekttyp: **Article**

Zeitschrift: **Bulletin.ch : Fachzeitschrift und Verbandsinformationen von Electrosuisse, VSE = revue spécialisée et informations des associations Electrosuisse, AES**

Band (Jahr): **99 (2008)**

Heft 5

PDF erstellt am: **28.05.2024**

Persistenter Link: https://doi.org/10.5169/seals-855827

Nutzungsbedingungen

Die ETH-Bibliothek ist Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Inhalten der Zeitschriften. Die Rechte liegen in der Regel bei den Herausgebern.
Die auf der Plattform e-periodica veröffentlichten Dokumente stehen für nicht-kommerzielle Zwecke in Lehre und Forschung sowie für die private Nutzung frei zur Verfügung. Einzelne Dateien oder Ausdrucke aus diesem Angebot können zusammen mit diesen Nutzungsbedingungen und den korrekten Herkunftsbezeichnungen weitergegeben werden.
Das Veröffentlichen von Bildern in Print- und Online-Publikationen ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Die systematische Speicherung von Teilen des elektronischen Angebots auf anderen Servern bedarf ebenfalls des schriftlichen Einverständnisses der Rechteinhaber.

Haftungsausschluss

Alle Angaben erfolgen ohne Gewähr für Vollständigkeit oder Richtigkeit. Es wird keine Haftung übernommen für Schäden durch die Verwendung von Informationen aus diesem Online-Angebot oder durch das Fehlen von Informationen. Dies gilt auch für Inhalte Dritter, die über dieses Angebot zugänglich sind.

Ein Dienst der *ETH-Bibliothek*
ETH Zürich, Rämistrasse 101, 8092 Zürich, Schweiz, www.library.ethz.ch

http://www.e-periodica.ch

Computernetzwerke gegen unbekannte Angriffe schützen

Massnahmen zur automatisierten Erkennung und Bekämpfung

Für die Absicherung von Computernetzwerken gegen Angriffe von innen und aussen gibt es heute viele unterschiedliche Strategien und Produkte. Die meisten von Ihnen bieten allerdings nur wenig Schutz vor Angriffen, die eine bisher unbekannte Schwachstelle ausnutzen. Das EU-Projekt NoAH zeigt, wie neue Lösungsansätze aus der Forschung helfen können, diesem Problem zu begegnen.

Bernhard Tellenbach, Daniela Brauckhoff, Bernhard Plattner

Wie verheerend die Auswirkungen eines Angriffs auf die IT-Infrastruktur von Firmen und Privatpersonen sein können, stellten spätestens die grossflächigen Angriffe durch Würmer wie Blaster, SQLSlammer oder Sobig unter Beweis. Dass seit 2005 kaum mehr Angriffe mit ähnlichem Ausmass beobachtet worden sind, ist neben verbesserten Schutzmassnahmen auch auf den damit verbundenen Paradigmenwechsel bei den Angreifern zurückzuführen. Aufgrund der besseren Schutzmechanismen ist der Zeitbedarf zur Entwicklung eines neuen Angriffs oder Angriffswerkzeugs so stark angestiegen, dass eine Professionalisierung stattgefunden hat. Der Angreifer möchte einen möglichst grossen wirtschaftlichen Nutzen haben. Die hohe Aufmerksamkeit, die einem grossflächigen Angriff zuteil wird, wäre diesem Ziel abträglich. Trotz verminderter Medienpräsenz, verbesserten Schutzmassnahmen und rascherer Behebung von Sicherheitslücken bleibt aber das Risiko eines Angriffs hoch. Wie aber soll man sich gegen solche Angriffe schützen?

Signaturen gegen Angriffe

Schutzmassnahmen wie Intrusion Detection Systems (IDS) oder Virenscanner verwenden zur Erkennung eines Angriffs meist Signaturen, die manuell oder halbautomatisch erstellt werden. Dies hat zwei grundlegende Nachteile: Erstens muss für die Erkennung und Bekämpfung des Angriffs bereits eine passende Signatur vorhanden sein. Damit dies überhaupt möglich ist, muss der jeweilige Angriff oder zumindest ein Teil der dabei verwendeten Methoden bekannt sein.

Zweitens erfordert die manuelle oder halbautomatische Signaturgenerierung viel Zeit. In der Zeit von der Entdeckung eines neuen Angriffs bis zur Verteilung und Installation der neuen Signatur bleiben die Systeme verwundbar. In der Vergangenheit haben dem Slammer-Wurm 10 Minuten ausgereicht, um 90% aller verwundbaren Computer im Internet zu infizieren [1]. Wenn der Schutz gegen neue Angriffe verbessert werden soll, müssen also die heutigen Schutzmassnahmen durch neue ergänzt werden. Ein vielversprechender Lösungsansatz, der neue und bereits bestehende signaturbasierte Massnahmen kombiniert, müsste Folgendes leisten:

- Erkennen von neuen Angriffen vor dem ersten erfolgreichen Angriff.
- Generierung einer passenden Signatur vor dem ersten erfolgreichen Angriff.

Wie ein erster grosser Schritt in diese Richtung aussehen kann, zeigt das im April 2005 und noch bis September 2008 laufende EU-Projekt Network of Affined Honeypots (NoAH) [2]. An diesem Projekt sind 8 Partner aus Forschung und Industrie, darunter auch die ETH Zürich, beteiligt. Das Projekt basiert auf der Verwendung von Honeypots. Diese werden eingesetzt, um Angriffe bereits vor dem ersten erfolgreichen Angriff auf Produktivsysteme aufzudecken.

Honeypots

Ein Honeypot ist im Prinzip nichts anderes als eine Falle für nicht legitimierte Benutzer oder Angreifer. Ein Honeypot ist normalerweise ein Computer, auf dem ein Abbild eines gewöhnlichen Computers oder

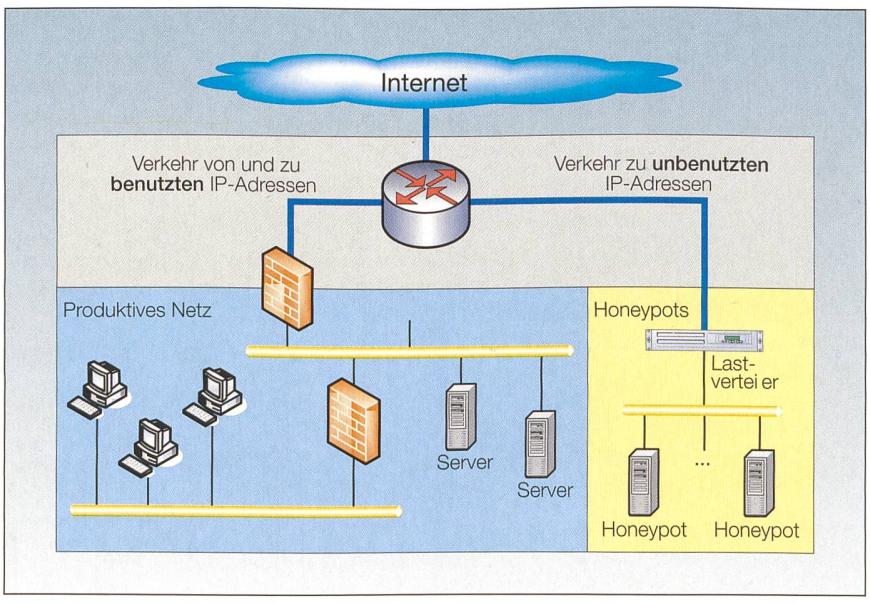

Bild 1 Firmennetzwerk mit Honeypots im unbenutzten IP-Adressbereich.

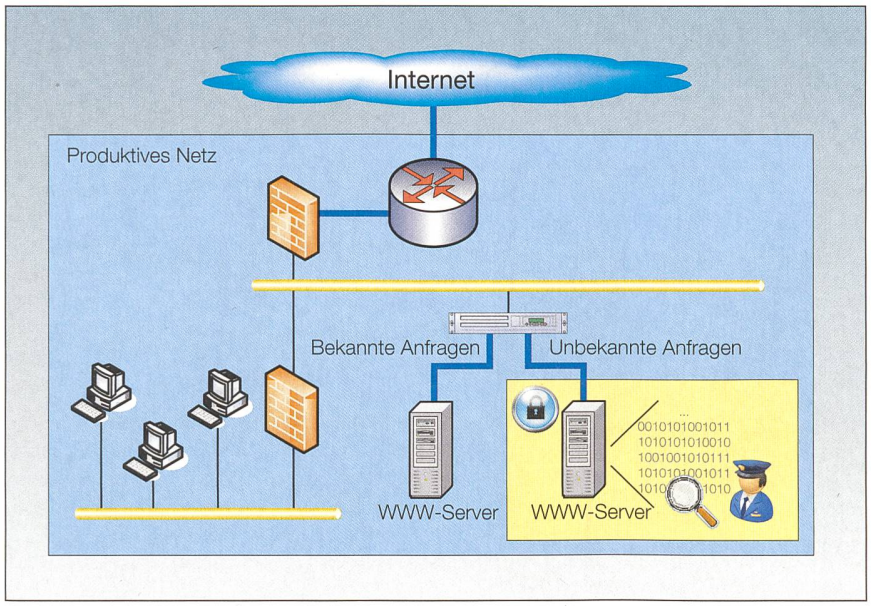

Bild 2 Firmennetzwerk mit einem Shadow-Honeypot.

auch eines Servers läuft. Im Unterschied zum normalen Computer oder Server werden die Honeypots aber nicht als solche genutzt. Aus diesem Grund gibt es keine legitime Nutzung des Honeypots. Wird trotzdem eine Nutzung registriert, ist diese per Definition illegitim. Die Aufzeichnung der Aktivitäten auf dem Honeypot ermöglicht es somit, Informationen über Angreifer und Angriffstechniken zu sammeln. Damit die Chance, dass ein Angreifer zuerst auf dem Honeypot landet, möglichst gross ist, sollte die Anzahl der Honeypots ebenfalls möglichst gross sein. Eine Möglichkeit, dies auf wirtschaftliche Art und Weise zu erreichen, ist der Einsatz einer Router/Lastverteiler-Kombination. Bild 1 zeigt ein Beispiel eines solchen Systemaufbaus innerhalb eines Firmennetzwerks. Die Router/Lastverteiler-Kombination verteilt jeglichen Verkehr zu nicht verwendeten IP-Adressen auf einige wenige Honeypots. Geht man davon aus, dass ein Angreifer sein Angriffsziel zufällig aus dem Adressbereich der Firma auswählt, dann landete er bei gleich viel benutzten wie unbenutzten IP-Adressen mit einer Chance von 50% zuerst auf einem Honeypot.

Ein weiteres Mittel, einen Angriff vor dem ersten erfolgreichen Angriff zu entdecken, ist der Einsatz von sogenannten Shadow-Honeypots (Bild 2). Im Gegensatz zu einem normalen Honeypot ist der Shadow-Honeypot auf den Schutz eines spezifischen Serverdienstes ausgerichtet. Der Einsatz erfolgt so, dass ein Vorfilter bereits bekannte Anfragen oder Anfragen mit bekanntem Format an den echten Server und unbekannte an den Shadow-Honeypot weiterleitet. Dieser bearbeitet dann die Anfrage und fügt sie zu den unbedenklichen Anfragen hinzu, falls kein Alarm ausgelöst wurde. Der Grund, dass die Detektionsmechanismen nicht direkt auf dem echten Server implementiert werden, ist auf den damit verbundenen hohen Performanceverlust zurückzuführen.

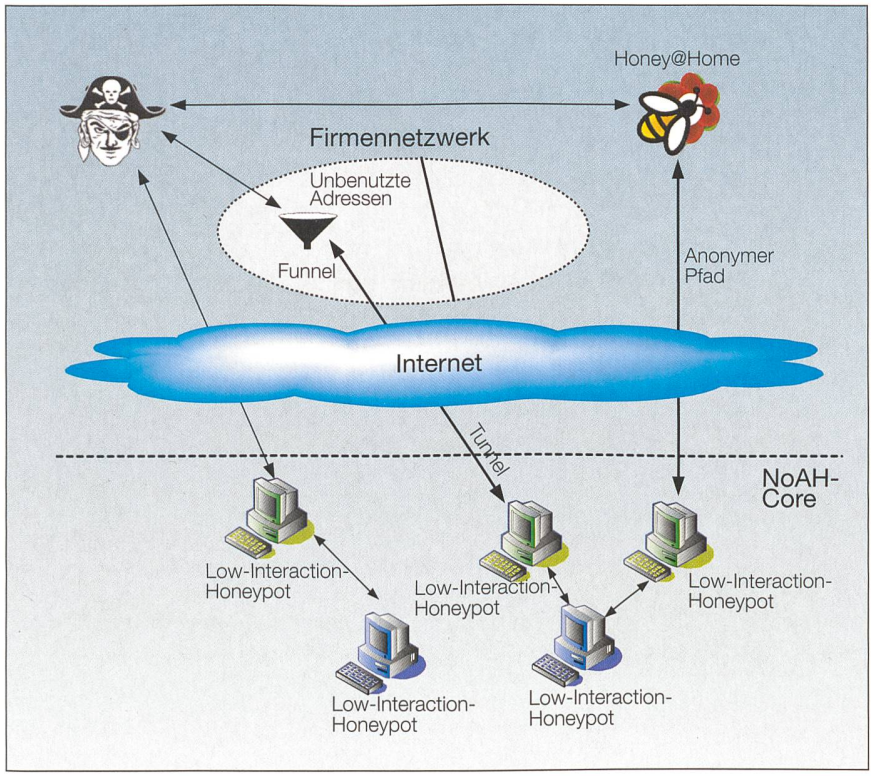

Bild 3 Basiskomponenten und Struktur der NoAH-Architektur.

Neue Angriffe abfangen

Um unbekannte Angriffe möglichst vor dem ersten erfolgreichen Angriff abfangen zu können, setzt NoAH primär auf eine möglichst gute Abdeckung des im Internet zur Verfügung stehenden Adressraums durch Honeypots. Der Einsatz von Shadow-Honeypots ist hierbei nur zum Schutz von ausgewählten Webservern vorgesehen.

Das Erreichen einer guten Abdeckung des Adressraums erfordert allerdings die Einbindung von Adressraum, der sich im Besitz von Drittpersonen befindet. Neben Aspekten wie der Skalierbarkeit bezüglich des Verkehrsaufkommens stellt dies die grösste Herausforderung an eine geeignete Architektur für NoAH dar.

Die NoAH-Architektur

Bild 3 zeigt die von NoAH vorgeschlagene Lösung. Die drei Komponenten, die für eine gute Abdeckung des Adressraums sorgen sollen, sind:

- Forwarding-Tunnel (Funnel): Ein einfaches Gerät, das den Verkehr an bestimmte Adressen verpackt und direkt den NoAH-Cores weiterleitet.
- Honey@Home: Software, die den Verkehr zu einer unbenutzten Adresse auf anonyme Art und Weise weiterleitet.
- NoAH-eigene Adressen: Angriffe auf Adressen der NoAH-Cores.

Die Konzentration der Honeypots im NoAH-Core ermöglicht es, die Komplexität aufseiten von Drittpersonen zu minimieren

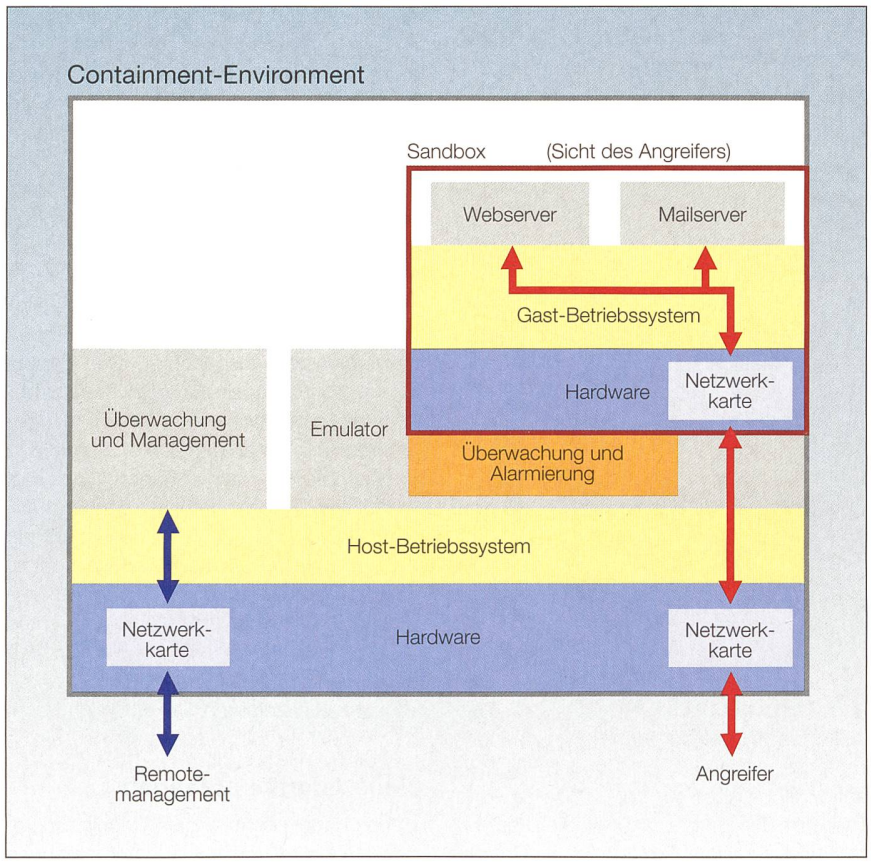

Bild 4 Aufbau des High-Interaction-Honeypots.

und das Management zu zentralisieren. Weiter erlaubt das anonyme Weiterleiten des Verkehrs von Honey@Home, auch Drittpersonen einzubinden, bei denen nicht bekannt ist, ob sie vertrauenswürdig sind. Würde der Verkehr nicht anonym weitergeleitet, könnte ein Angreifer Honey@Home benutzen, um die Adressen der NoAH-Cores zu identifizieren. Er könnte dann möglicherweise durch eine gezielte Denial-of-Service-Attacke (DoS) verhindern, dass NoAH seinen neuen Angriff abfängt und Alarm schlägt.

Der NoAH-Core

Der ebenfalls im Bild 3 zu sehende NoAH-Core erfüllt die Kernaufgabe von NoAH: Die Erkennung von noch unbekannten Angriffen. Der Kern besteht hauptsächlich aus einer Reihe von Low- und High-Interaction-Honeypots. Die Low-Interaction-Honeypots sind Maschinen, die zur Kommunikation mit dem Angreifer nicht die echten Programme verwenden, sondern nur deren Kommunikationsverhalten emulieren. Dies ermöglicht ihnen, mehrere Computer mit den entsprechenden echten Programmen zu ersetzen. In NoAH dienen diese Low-Interaction-Honeypots der Vorfilterung des ankommenden Verkehrs. Die Low-Interaction-Honeypots sollen auf Angriffe antworten, die bereits bekannt sind. Sobald aber ein unbekanntes Kommunikationsmuster erkannt wird, übergibt er die Kommunikation einem High-Interaction-Honeypot. Auf diesem kommuniziert der Angreifer dann mit dem echten Programm. Durch diese Vorfilterung kommt der NoAH-Core auch mit einem hohen, durch bekannte Angriffe verursachten Verkehrsvolumen zurecht.

Wie aber erkennt der High-Interaction-Honeypot, ob ein an ihn weitergeleiteter Angriff überhaupt erfolgreich ist und dadurch eine echte Gefahr für andere darstellt? NoAH setzt hier auf eine neu entwickelte Technik, die das Einschleusen und Ausführen von fremdem Code erkennen und verhindern kann. Beispiele für diese Technik sind TaintCheck [3] und das von einem NoAH-Partner entwickelte Argos [4]. Das Funktionsprinzip dieser Technik basiert einerseits auf dem Sandboxing und andererseits auf dem Memory-Tainting-Prinzip.

Sandboxing

Unter Sandboxing versteht man, dass Programme oder auch Betriebssysteme mit zugehörigen Programmen in einer isolierten Umgebung (Sandbox) laufen. Die Sandbox verhindert dabei jeglichen schädlichen Einfluss auf die Umgebung, in der die Sandbox läuft. Gleichzeitig erlaubt sie eine genaue Beobachtung der darin ablaufenden Programme, ohne dass diese die Sandbox erkennen. Die Umgebung, in der die Sandbox läuft, wird oft auch Containment Environment genannt. Bild 4 illustriert den Einsatz von Argos, dem Containment Environment des NoAH-Projekts. Die Sandbox wird hier durch Emulierung eines ganzen Computers in Software realisiert. Von aussen angreifbar

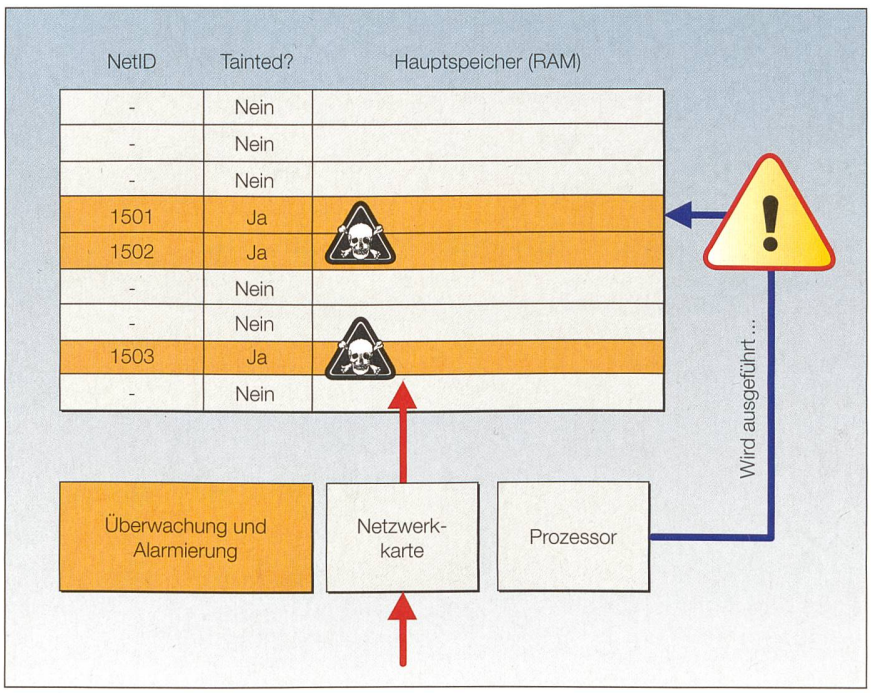

Bild 5 Markieren von potenziell gefährlichen Daten (Memory Tainting).

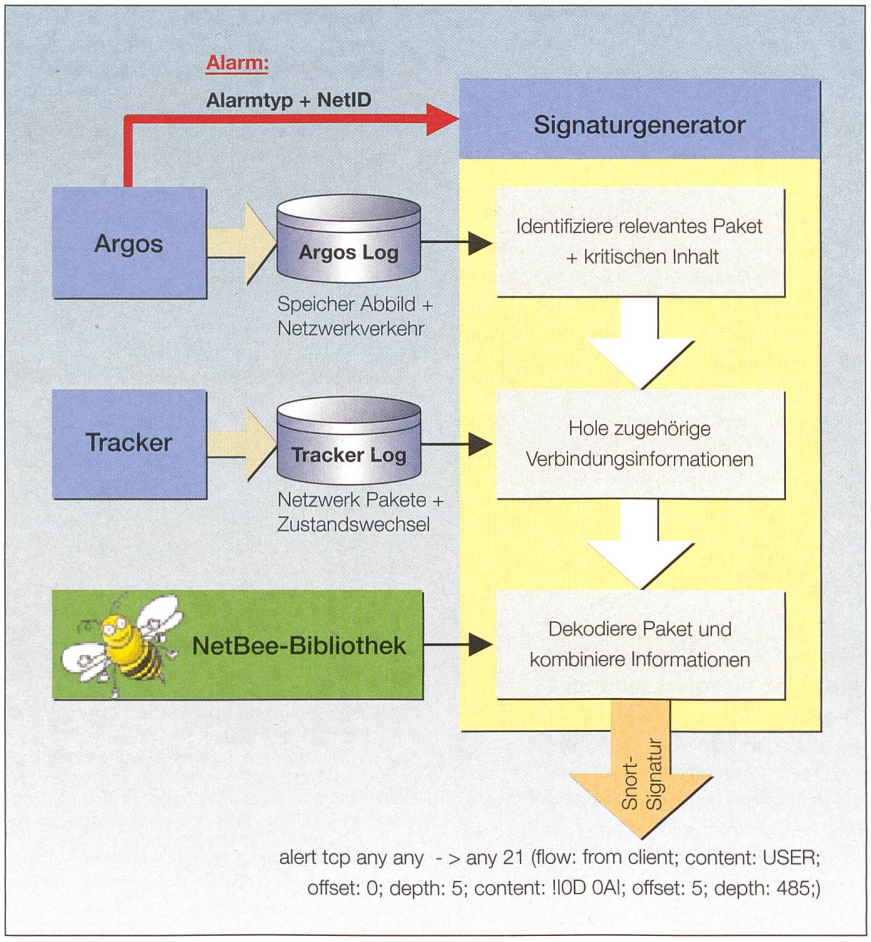

Bild 6 Funktionsweise der automatischen Signaturgenerierung.

ist einzig und alleine diejenige Software, die innerhalb des emulierten Computers (Sandbox) läuft. Dies wird durch das Weiterleiten des eingehenden Netzwerkverkehrs an die Netzwerkkarte von Argos erreicht. Der Angreifer sieht dadurch einzig und alleine den emulierten Computer, ohne zu wissen, dass es sich um einen solchen handelt. Der Grund, wieso NoAH gerade diese Art von Sandboxing einsetzt, liegt im dadurch ermöglichten Einsatz von betriebssystemunabhängigem Memory Tainting. Diese für NoAH zentrale Technik ermöglicht es, alle bekannten und unbekannten Angriffe, die auf der Einschleusung und Ausführung von Code beruhen, zu erkennen.

Memory Tainting

Damit das Containment Environment erkennt, wann es ein Angreifer geschafft hat, eigenen Code einzuschleusen und zur Ausführung zu bringen, werden alle über die Netzwerkkarte hereinkommenden Daten markiert (getainted), und deren Position wird innerhalb des empfangenen Netzwerkverkehrs aufgezeichnet (Bild 5). Argos überwacht fortan alle Operationen, in die markierte Daten involviert sind, und vererbt die Markierung gegebenenfalls an daraus hervorgehende neue Daten weiter. Wenn Argos feststellen sollte, dass der emulierte Prozessor plötzlich Programmcode von einer markierten Speicherstelle ausführen soll, hat es ein Angreifer geschafft, eigenen Code einzuschleusen und zur Ausführung zu bringen. Unter Ausnutzung welcher (unbekannten) Schwachstelle er das geschafft hat, ist für die Erkennung irrelevant. Sie ist für alle Angriffe, wo der Angreifer eigenen Code zur Ausführung bringt, garantiert.

Automatische Signaturgenerierung

Um eine passende Signatur noch vor dem ersten erfolgreichen Angriff generieren zu können, setzt NoAH nicht auf die konventionelle manuelle oder halbautomatische Signaturgenerierung, sondern auf ein eigens entwickeltes Verfahren zur automatischen Signaturgenerierung. Dieses startet, sobald Argos einen Alarm generiert. Von der Erkennung des Angriffs bis zur Bereitstellung der fertigen Signatur im Snort-Format vergehen nur wenige Sekunden. Die dabei erzeugte Signatur kann direkt im Snort IDS [5] eingesetzt werden, um verwundbare Systeme zu schützen.

Damit die Signatur keine Fehlalarme generiert oder Angriffe übersieht, muss sie die ausgenutzte Schwachstelle so genau wie möglich erfassen. NoAH greift dabei auf eine Kombination aus verschiedenen Techniken zurück, unter anderem auf das Wissen über die Struktur von Kommunikationsprotokollen. Die Signaturgenerierung läuft im Wesentlichen in vier Schritten ab (siehe auch Bild 6): Im ersten Schritt identifiziert der Signaturgenerator das Netzwerkpaket sowie die Ursache und die genaue Position des kritischen Inhalts, der gemäss Argos für den Alarm verantwortlich war. Dazu greift er auf Informationen aus den Log-Dateien von Argos zurück.

In einem zweiten Schritt wird die Vorgeschichte zu diesem Paket analysiert, und die durchlaufenen Zustände für die unterschiedlichen Protokollschichten werden festgehalten. Die Komponente, die diese Analyse ermöglicht, ist die Tracker-Komponente. Durch Mithören auf der Netzwerkverbindung des Honeypots registriert sie alle Zustandswechsel und Pakete, die der Honeypot empfängt oder sendet. Falls das Signaturformat dies erlaubt, kann die Signatur so formuliert werden, dass nur Pakete, die in genau diesem Zustand (oder mit einer bestimmten Vorgeschichte an Zustandswechseln) empfangen wurden, weiter ausgewertet werden.

Im dritten Schritt wird auf die NetBee-Bibliothek [6] zugegriffen. NetBee ist eine Open-Source-Bibliothek, die unter anderem die Dekodierung von Netzwerkpaketen vornehmen kann. Das Ergebnis der Dekodierung ist die Kenntnis der Grenzen der einzelnen Informationsfelder. Mit dieser Information kombiniert mit der Positionsinformation des kritischen Inhalts aus Schritt 1, kann das verantwortliche Informationsfeld identifiziert werden. Die Signatur wird nun so formuliert, dass sie nur genau diese Informationsfelder weiter auswertet. Sollte das Protokoll, über das der Angriff läuft, von der Bibliothek nicht unterstützt werden, lässt NoAH den nächsten Schritt aus und generiert eine Signatur, die auf dem Bytemuster des kritischen Inhalts basiert.

Im letzten Schritt spielt die Ursache des Alarms eine Rolle. Für Angriffe, die beispielsweise auf einem einfachen Buffer-Overflow basieren, sind die vom Signaturgenerator generierten Signaturen perfekt. Dies wird dadurch erreicht, dass die Signatur dann einen Alarm auslöst, wenn das im dritten Schritt identifizierte Informationsfeld gleich viele oder mehr Daten enthält als von Beginn des Feldes bis zum Beginn der kritischen Daten. Im Gegensatz zu einer

Signatur, die auf einem bestimmten Bytemuster basiert, ist diese Signatur unabhängig vom Inhalt des Feldes.

Bis 50 Angriffe pro Tag

Ein Testaufbau, basierend auf einer kleinen Anzahl von Honeypots, verdeutlicht, dass wir unter permanentem Beschuss durch ungewollten und meist bösartigen Verkehr stehen. Ein Honeypot wird täglich ca. 20- bis 50-mal angegriffen. Dies gilt pro IP-Adresse, denn ein physikalischer Rechner als Honeypot kann für mehrere IP-Adressen zuständig sein. Ein Angriff ist in diesem Fall eine Verbindung, die effektiv eine Schwachstelle ausgenutzt hat. Die genaue Anzahl hängt ziemlich stark davon ab, in welchem Netzwerk die IP-Adresse steht. Ist ein produktives Netzwerk angrenzend, ist die Anzahl der registrierten Angriffe deutlich höher. Weiterhin schwankt die Anzahl der Angriffe mit der Uhrzeit. Die meisten Angriffe sehen wir zwischen 12 und 20 Uhr.

Das NoAH-Projekt hat Wege aufgezeigt, wie unbekannte Angriffe frühzeitig erkannt und mithilfe der automatischen Signaturgenerierung auch schnell bekämpft werden können. Es gilt nun, den eingeschlagenen Weg weiterzugehen und die eingesetzten Techniken zu verfeinern und auszubauen. Wird dies konsequent verfolgt, könnte NoAH bald in einem Produkt münden, das die bestehenden Schutzmechanismen ideal ergänzt.

Résumé

Protection des réseaux d'ordinateurs contre les attaques inconnues

Mesures d'identification et de lutte automatiques. Il existe actuellement de nombreux produits et stratégies destinés à protéger les réseaux d'ordinateurs des attaques de l'intérieur et de l'extérieur. Cependant, la plupart d'entre eux n'offrent que peu de protection contre les attaques profitant d'une faiblesse encore inconnue. Le projet NoAH de l'UE montre comment de nouvelles ébauches de solution fournies par la recherche permettent de s'attaquer au problème.

Referenzen

[1] D. Moore et al.: Inside the Slammer worm. IEEE Security and Privacy 1,4 (Juli 2003).
[2] NoAH-Projekt Webseite: www.fp6-noah.org.
[3] J. Newsome, D. Song: Dynamic Taint Analysis for Automatic Detection, Analysis, and Signature Generation of Exploits on Commodity Software. NDSS 2005.
[4] G. Portokalidis, A. Slowinska, H. Bos: Argos – An Emulator for Fingerprinting Zero-Day Attacks. ACM SIGOPS Eurosys 2006.
[5] Snort IDS: http://www.snort.org/.
[6] NetBee: www.nbee.org/.

Angaben zu den Autoren

Bernhard Tellenbach arbeitet am Institut für Technische Informatik und Kommunikationsnetze (TIK) der ETH Zürich. Sein Forschungsschwerpunkt liegt im Bereich der Computer- und Netzwerksicherheit. Ausserdem ist er als selbstständiger Berater im Bereich der IT-Sicherheit sowie seit 2006 als Lehrbeauftragter an der Hochschule für Technik Rapperswil tätig.
TIK, ETH Zürich, 8092 Zürich,
tellenbach@tik.ee.ethz.ch

Daniela Brauckhoff arbeitet am Institut für Technische Informatik und Kommunikationsnetze (TIK) der ETH Zürich. Ihr Forschungsschwerpunkt liegt im Bereich der Computer- und Netzwerksicherheit. Ausserdem ist sie als selbstständige Beraterin im Bereich der IT-Sicherheit tätig.
TIK, ETH Zürich, 8092 Zürich,
brauckhoff@tik.ee.ethz.ch

Prof. Dr. *Bernhard Plattner* ist ordentlicher Professor für Technische Informatik und Leiter der Communication Systems Group an der ETH Zürich. Von 2005 bis 2007 war er Vizerektor für die Bachelor- und Master-Studiengänge.
TIK, ETH Zürich, 8092 Zürich,
plattner@tik.ee.ethz.ch