Honeypots und Honeynets

R. Stevens · H. Pohl

Honeypots sind Server mit nur scheinbar wertvollen Daten wie Adressen und Dokumenten zur Täuschung von Angreifern [14]. Mit ihnen soll von Systemen abgelenkt werden, die tatsächlich wertvolle Daten verarbeiten.

Zusätzlich werden die Angriffe auf diese Honeypots beobachtet und analysiert, um neuartige Angriffe kennen zu lernen; dazu werden sie protokolliert und ggf. auch rückverfolgt. Angriffe auf Honeypots sind also erwünscht.

Unter einem Honeynet wird eine Menge vernetzter Honeypots mit denselben Aufgaben und Zielen verstanden. Der Vorteil eines Honeynets gegenüber einem Honeypot liegt in der Überwachungs- und Verwaltungsmöglichkeit der einzelnen Honeypots von einem anderen System aus. Damit lassen Honeynets auch die Beobachtung erfolgreich angegriffener Systeme zu, auf denen sich der Angreifer volle Administrationsrechte verschafft hat: Die schützende und überwachende Infrastruktur liegt nicht auf dem einzigen Honeypot selbst, sondern außerhalb der Zugriffsmöglichkeiten des Angreifers. Dies verursacht allerdings einen höheren Aufwand für Konzeption, Implementierung und Betrieb.

Honeypots und Honeynets wurden bisher überwiegend für Forschungszwecke eingesetzt. Der höhere Aufwand, die größere Komplexität und die Möglichkeit, Angriffsverfahren zu erkennen, scheinen den Einsatz von Honeynets in Unternehmen nicht zu rechtfertigen.

Neuere Untersuchungen [17] zeigen allerdings, dass ein längerfristiger Einsatz von Honeynets in Unternehmen auch Informationen liefern kann, die die Sicherheit des Unternehmensnetzwerks direkt – zumindest aber indirekt – verbessern können. Ein erkannter und beobachteter Angriff ermöglicht sofortige Reaktion und liefert daher einen direkten Nutzen. Die statistische Auswertung der Zugriffe auf das Honeynet kann indirekt genutzt werden, um andere Sicherheitsmechanismen anzupassen und liefert damit indirekten Nutzen. Ein Honeynet bietet zudem eine kontrollierte Testumgebung, in der neue Systeme vorab oder parallel zur Einführung betrieben und genau beobachtet werden können.

Derzeitige Angriffe sind wie folgt in mehreren Stufen aufgebaut [5]: Bevor überhaupt ein Angriff erfolgt, wird das Ziel aufgeklärt (Footprinting) und ermittelt, welche Netzsegmente und Maschinen zum Angriffsziel gehören sollen. Dann folgt ein Scannen, um herauszufinden, über welche Ports Kommunikationsverbindungen aufgebaut werden können. Danach wird geprüft, welches Betriebssystem auf den Zielrechnern eingesetzt wird und ob – und gegebenenfalls sogar welche – Firewalls die Systeme schützen. In der letzten Stufe der Vorbereitung ermittelt der Angreifer die Versionen der für ihn sichtbaren Serverdienste (Bannergrabbing). Nachdem das Angriffsziel analysiert wurde, erfolgt der Angriff z. B. durch

DOI 10.1007/s00287-004-0404-y

R. Stevens
Accenture GmbH,
Post & Public Services,
Düsseldorf

Prof. Dr. H. Pohl
Fachbereich Informatik,
Fachhochschule Bonn-Rhein-Sieg,
ISIS InStitut für InformationsSicherheit,
Max-Pechstein-Str. 4, 50858 Köln
E-Mail: Hartmut.Pohl@sang.net

* Vorschläge an Prof. Dr. Frank Puppe <puppe@informatik.uni-wuerzburg.de> oder Dieter Steinbauer <dieter.steinbauer@schufa.de>

Alle „Aktuellen Schlagwörter" seit 1988 finden Sie unter: www.ai-wuerzburg.de/as

Ausnutzen bekannter Sicherheitslücken in einem der erkannten Dienste.

Honeypots

Honeypots sollen angegriffen werden, damit die Angriffe analysiert werden können; Honeypots übernehmen keine produktiven Aufgaben. Daraus resultiert, dass ein in einem Netzwerk platzierter Honeypot, im Vergleich zu produktiven Systemen (fast) keine berechtigten Zugriffe aufweist. Jeder Zugriff wird als verdächtig angesehen und untersucht. Die vergleichsweise geringe Menge gespeicherter (scheinbarer) Nutzdaten erlaubt leichter eine vollständigere Protokollierung, weil die Zugriffe im Gegensatz zu produktiven Systemen nicht in der Masse der berechtigten Zugriffe untergehen.

Bei Verdacht auf einen Angriff wird vom Honeypot Alarm ausgelöst und eine gezielte Beobachtung des Angreifers und des Angriffsverfahrens vorgenommen – noch bevor der Angriff erfolgreich ist – mindestens aber während des Angriffs. Eine Analyse neuer – durch einen Honeypot erkannter – Angriffe kann zur Erstellung von Regeln für ein Intrusion Detection oder Intrusion Protection System genutzt werden.

Auf Produktionssystemen müssten verdächtige Zugriffe dagegen aus der Vielzahl der berechtigten Zugriffe – z. B. mit Intrusion-Detection-Systemen – erst herausgefiltert werden. Intrusion-Detection-Systeme untersuchen die Datenströme im Netzwerk und vergleichen sie mit den Signaturen ihnen bekannter Angriffe. Wird eine Übereinstimmung festgestellt, generiert das System einen Alarm. Dieses Verfahren birgt einige konzeptbedingte Probleme wie die ausschließliche Erkennung nur bekannter Angriffsmuster und Fehlalarme bei Angriffsmustern ähnelnden berechtigten Zugriffen.

Klassifikation nach Zielen

Mit dem Einsatz von Honeypots können zwei Ziele verfolgt werden [15]: Produktiver Einsatz zum Schutz der IT oder Forschungszwecke wie Kennenlernen von Angriffsmethoden, -werkzeugen und -taktiken.

Honeypots sollen für Angreifer möglichst interessant erscheinen (nur scheinbar wertvolle Nutzdaten: Täuschung) und den Angreifer vom Angriff auf tatsächlich wertvolle Daten verarbeitende Systeme ablenken.

Produktionshoneypots (Production Honeypots)

Produktionshoneypots melden (unberechtigte) Zugriffe. Die Zugriffe können entweder auf Fehler oder auf Angreifer zurückgeführt werden; jedenfalls werden diese Zugriffe als Angriffe behandelt und es wird ein Alarm ausgelöst. In der Folge des Alarms lassen sich die tatsächlichen Produktionssysteme gegen den geführten Angriff absichern.

Forschungshoneypots (Research Honeypots)

Forschungshoneypots sollen möglichst viel über Angriffsmethoden und die eingesetzten Werkzeuge in Erfahrung bringen. Die detaillierte Ausprägung dieser Systeme ist abhängig von den Zielen.

Sollen z. B. neue Würmer analysiert werden, werden entsprechende Sicherheitslücken offen gelassenen. Nach einer Infektion des Systems kann der Wurm und seine Funktionsweise analysiert werden.

Eine weiteres im Forschungsbereich angesiedeltes Einsatzfeld ist die Trenderkennung: An verschiedenen Stellen im Internet platzierte Honeypots können Daten über die Zugriffsversuche auf verschiedene Ports liefern. Die Zugriffshäufigkeiten können Hinweise auf neue Angriffswerkzeuge liefern.

Die Ergebnisse werden unter Betreibern im Rahmen von Vereinigungen ausgetauscht [4, 15].

Klassifikation nach Interaktionsgrad

Honeypots können auch nach Interaktionsmöglichkeiten klassifiziert werden, die sie Angreifern bieten. Drei Klassen werden unterschieden [15].

Honeypots mit niedrigem Interaktionsgrad („low interaction")

Einfache Programme simulieren Dienste, ohne dass sie nutzbar oder überhaupt vorhanden sind. Ein simulierter Telnet-Dienst könnte z. B. einen Login-Prompt gefolgt von einem Passwort-Prompt anbieten – jedoch bei Anmeldeversuchen immer eine Fehlermeldung zurückliefern. Die Klasse der Honeypots mit geringem Interaktionsgrad eignet sich zur Angriffserkennung und Trenderkennung. Es können jedoch keine Informationen gesammelt werden, die über Verbindungsversuche und deren Häufigkeit hinausgehen.

Produktbeispiele für Honeypots mit niedrigem Interaktionsgrad sind Specter [12] und Honeyd [13]. Die von Specter angebotenen Dienste können über

Checkboxen ausgewählt und aktiviert werden. Dadurch kann ein Specter-Honeypot mit wenig Aufwand implementiert werden. Honeyd kann eher als ein Werkzeugkasten für Honeypots betrachtet werden. Dienste werden über Skripte simuliert, die bei Zugriff auf die konfigurierten Netzwerk-Ports durch den Honeyd gestartet werden. Diese Eigenschaft verursacht im Vergleich zu Specter etwas mehr Aufwand, erhöht jedoch auch die Flexibilität.

Honeypots mit mittlerem Interaktionsgrad („medium interaction")

Diese simulieren angebotene Dienste. Sie sind besonders für das Beobachten programmgesteuerter Angriffswerkzeuge nützlich, da diese im Gegensatz zu menschlichen Angreifern nur die einprogrammierten Eigenschaften der Dienste prüfen und dann eine Schadensfunktion ausführen.

Ein Beispiel für einen Honeypot mit mittlerem Interaktionsgrad ist ein System, dass eine bekannte Sicherheitslücke eines Webservers nachbildet; Würmer, die diese Lücke ausnutzen, sind zwar in der Lage, ihre Schadensroutine zu installieren, können sie aber nicht aktivieren. Das System kann einen Alarm auslösen, wenn eine bisher nicht bekannte Schadensroutine abgeliefert wird. Diese kann dann analysiert werden. Honeypots mittleren Interaktionsgrads müssen speziell auf ihren Einsatzzweck hin entwickelt werden.

Zur Implementierung von Honeypots mit mittlerem Interaktionsgrad kann auch Honeyd eingesetzt werden, da über entsprechend umfangreichere Skripte auch komplexere Dienste simuliert werden können.

Honeypots mit hohem Interaktionsgrad („high interaction")

Diese Honeypots stellen einem Angreifer den vollen Funktionsumfang eines Betriebssystems zur Verfügung. Meist handelt es sich um Standard-Betriebssysteminstallationen, die entsprechend modifiziert wurden: Auf UNIX-artigen Systemen wird häufig ein Keylogger installiert, der sämtliche Ein- und Ausgaben aller Terminals protokolliert. Darüber hinaus werden so viele Protokollinformationen wie möglich an entfernte Rechner verschickt; damit ist die Protokollinformation auch dann noch verfügbar, wenn ein Angreifer sie auf dem Zielrechner gelöscht hat – z. B. um seine Spuren zu verwischen. Es sollen möglichst viele Informationen protokolliert werden, ohne dass ein Angreifer davon etwas merkt oder sogar die protokollierten Informationen zerstören könnte.

Da Betriebssysteme mit vollwertigen Diensten eingesetzt werden, bieten diese Honeypots eine größere Angriffsfläche. Über bewusst offengelassene oder bisher unbekannte Sicherheitslücken in der Dienstesoftware oder der Konfiguration des Systems können Angreifer daher auf diesen Honeypots Zugriffsrechte oder sogar Administrationsrechte erlangen. Zwar ist einerseits eine genauere Beobachtung der Angriffe möglich, andererseits bietet ein vollständiges Betriebssystem dem Angreifer mehr Missbrauchsmöglichkeiten.

Auf nicht besonders geschützten Systemen könnte ein Angreifer mit Administrationsrechten die Überwachungsmechanismen erkennen, deaktivieren und damit den Nutzen des Honeypots einschränken. Ein Angreifer mit Administrationsrechten ist auf den meisten heute üblichen Betriebssystemen nur schwer kontrollierbar. Ausnahmen bilden gehärtete Systeme [8, 9, 10], die die Rechte der Prozesse und Benutzer durch Mandatory Access Control reduzieren können. SELinux [11] ist z. B. eine Erweiterung für Linuxsysteme, die Einschränkungen möglich macht.

Honeynets

Um auch bei nicht besonders abgesicherten Systemen Kontrolle und Schutz der Protokolldaten gewährleisten zu können, wurden Honeynets entwickelt. Durch die Honeynet-Infrastruktur können auch Systeme kontrolliert werden, auf denen ein Angreifer Administrationsrechte hat (Abb. 1).

Im Honeynet Project [16] werden drei Ziele definiert, die bei der Konzeption eines Honeynets berücksichtigt werden sollen.

- Die Kontrolle (Data Control) stellt sicher, dass ein erfolgreicher Angreifer das Zielsystem nicht für weitere Angriffe nutzen kann.
- Die Datenerfassung (Data Capture) soll eine möglichst vollständige Überwachung des Angreifers sicherstellen (Protokollierung). Kontrolle und Datenerfassung sollen von Angreifern nicht bemerkt werden.
- Alle gesammelten Daten sollen archiviert (Data Collection) und an einer zentralen Stelle zur Auswertung bereit gestellt werden – z. B. um zukünftig Daten verschiedener Honeynets an dieser zentralen Stelle zu sammeln und die Ergebnisse (besser) korrelieren zu können. Die Entwicklung von Konzepten und Werkzeugen für eine zentrale (gemeinsame)

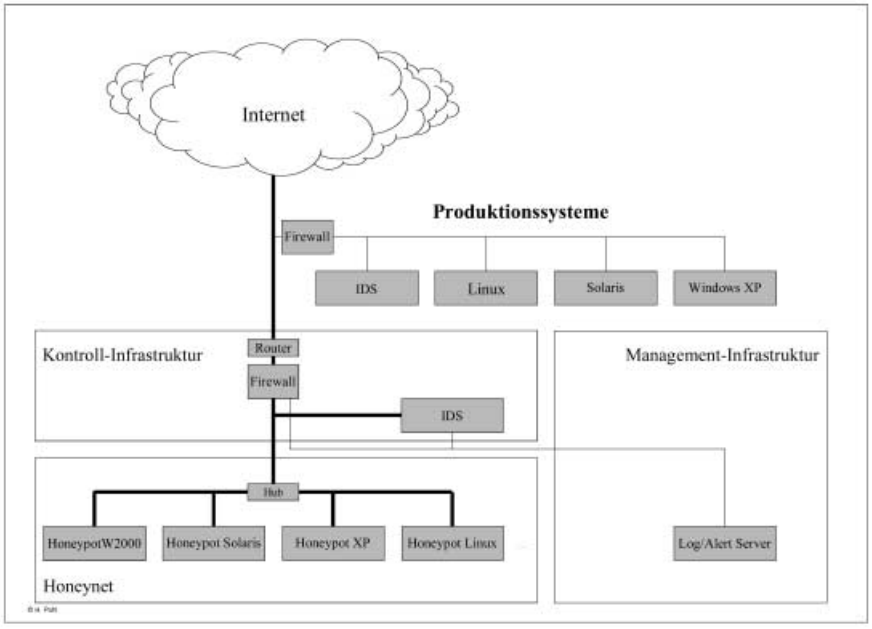

Abb. 1 Honeynet-Konfiguration

Datenhaltung und -auswertung ist eins der noch offenen Forschungsprojekte des Honeynet Projects.

Honeypots mit hohem Interaktionsgrad und Honeynets lassen sich durch Virtualisierung auch auf einem einzigen Rechner implementieren. Dabei kommen virtuelle Maschinen wie VMware Workstation oder GSX-Server [19] oder User-Mode Linux [18] zum Einsatz. In den virtuellen Maschinen lassen sich Betriebssysteme als Prozess eines Hostsystems starten. Hardwarezugriffe der Gastsysteme werden entweder an die Hardware des Gastsystems weitergereicht oder durch Software emuliert. Eine Platte in einem User-Mode-Linux-System besteht beispielsweise aus einer Datei im Hostsystem. Auf dem Hostsystem können mehrere User-Mode-Linux-Instanzen gestartet werden und über virtuelle Hubs ein Netzwerk bilden.

Die Virtualisierung ermöglicht eine umfangreichere Protokollierung. Die Anpassung der in virtuellen Maschinen laufenden Betriebssysteme mit Keyloggern kann entfallen, da die virtuelle Maschine, anders als reale Hardware, die Zugriffe auf virtuelle Hardware protokollieren kann. Im CoVirt Projekt [2] wird eine auf User-Mode Linux basierende virtuelle Maschine, ein aus Performancegründen angepasstes Linux und ein Abspielservice entwickelt, mit denen sich Aktivitäten innerhalb der virtuellen Maschine so vollständig protokollieren lassen, dass sie zur Analyse Instruktion für Instruktion wieder abspielen können.

Die aktuelle Honeynet-Generation des Honeynet Projects erreicht die geforderten Ziele durch mehrere Komponenten. Die Kontrolle wird durch eine, dem Honeynet vorgeschaltete, Firewall realisiert, die als Bridge zum Internet fungiert. Bei Bedarf kann dieser Bridge-Firewall noch ein Router mit gleichem Regelwerk vorgeschaltet werden, um Redundanz zu erreichen. Die Firewall lässt alle eingehenden Verbindungen zu, um einen uneingeschränkten Zugriff auf die Zielsysteme zu erreichen.

Ausgehende Verbindungen sind besonders verdächtig, da sie auf einen erfolgreich angegriffenen Honeypot schließen lassen. Die Firewall ist so konfiguriert, dass sie jegliche ein- und ausgehende Verbindungen protokolliert. Alle ausgehenden Verbindungen lösen einen Alarm aus und werden darüber hinaus in ihrer Anzahl beschränkt, um eine Nutzung erfolgreich angegriffener Systeme für Massenscans oder Denial of Service-Attacken zu verhindern. Weiterhin wird der ausgehende Datenstrom durch Snort-Inline [7] gegen das Regelwerk des Snort Intrusion Detection Systems [1] geprüft: Damit können Pakete bekannter Attacken fallengelassen und in Zukunft auch modifiziert durchgelassen werden.

Die Datenerfassung wird durch einen im Honeynet platzierten Sniffer realisiert, der den Datenfluss im Honeynet vollständig erfasst und zu Auswertungszwecken speichert. Im Honeynet Project kommt das Intrusion-Detection-System Snort zum Einsatz. Der Snort-Sensor ist so konfiguriert,

dass er neben den Alarmen auch ein vollständiges Binärlog und Sitzungen unverschlüsselter textbasierter Protokolle erstellt. Durch die Sitzungsprotokolle lassen sich beispielsweise eingegebene Befehle und resultierende Antworten von FTP- oder Telnetsitzungen überwachen.

Sniffer und Firewall können über ein separates Netzsegment mit einem Wartungssystem verbunden werden, das einen Zugriff auf die Infrastruktur über das Internet erlaubt. Mit Hilfe der Sitzungsprotokolle und auf den Zielsystemen installierten Keyloggern lassen sich die Aktivitäten eines Angreifers in Echtzeit beobachten.

Bewertung

Honeypots weisen (fast) keine berechtigten Zugriffe auf; personenbezogene Daten werden daher in Honeypot- oder Honeynet-Systemen im Rahmen der technischen Überwachung der Kommunikation nicht protokolliert, gespeichert und ausgewertet; erst recht nicht solche von Mitarbeitern. Vielmehr fallen nur Angriffsdaten an: Mit welchen Verfahren und auf welche Daten wurde wann und wie zugegriffen. Im Allgemeinen wird auch auf eine Rückverfolgung der Angreifer wegen des entstehenden Aufwands verzichtet; weniger aufwändig aber wirkungsvoller ist nämlich die Absicherung der eigenen Systeme gegen diese neuen Angriffe [5].

Honeypots und Honeynets erfordern einen hohen Analyseaufwand der Protokolldaten. Es kann erwartet werden, dass diese Analyse zukünftig programmgesteuert vorgenommen wird und die Ergebnisse – ebenfalls programmgesteuert – unmittelbar in das Regelwerk von Sicherheitstools wie Firewalls, Intrusion Detection und Intrusion Protection Systemen eingespeist werden. Erste Ansätze dazu lassen sich bei einem System erkennen [3], das – als ein der Firewall vorgeschaltetes System – bei Anfragen auf nichtexistente Dienste mit „Ködern“ antwortet. Werden diese „Köderinformationen“ daraufhin benutzt, wird das Regelwerk der Firewall so angepasst, dass der Nutzer der „Köder“ keinen Zugriff mehr auf die vorhandenen Dienste erlangen kann und es wird zusätzlich ein Alarm ausgelöst.

Literatur

1. Caswell, B.; Roesch, M.: Snort. The open source network intrusion detection system. o. O. 2004
2. Chen, P.M., Noble, B. et al.: CoVirt/ReVirt Ann Arbor 2004
3. ForeScout (ed.): ActiveScout. San Mateo 2004 http://www.forescout.com
4. La Bella, R.: Florida Honeynet Project. o.O. 2004 http://www.sfhn.net
5. Landesbeauftragter für den Datenschutz Niedersachen: Protokollierung. Orientierungshilfe und Checkliste. Hannover 2000 http://www.lfd.niedersachsen.de/master/0,,C27924_N13192_L20_D0_I560,00.html
6. McClure, S.; Scambray, J.; Kurz, G.: Hacking exposed. Berkeley 2003
7. N.N. (ed.): Project: snort_inline. o.O. 2004 http://sourceforge.net/projects/snort-inline
8. Microsoft Corp. (ed.): Microsoft Windows 2000 Security Hardening Guide. Redmond/WA 2003 http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/prodtech/windows/win2khg.asp
9. National Security Agency (NSA) (ed.): Guide to securing Microsoft Windows XP. Fort Meade 2002 http://www.nsa.gov/snac
10. National Security Agency (NSA) (ed.): Security recommendation guides. Fort Meade 2003ahttp://www.nsa.gov/snac/winy/guides/wxy-1.pdf
11. National Security Agency (NSA) (ed.): SELinux. Fort Meade 2003bhttp://www.nsa.gov/selinux
12. NETSEC, Network Security Software (ed.): Specter. Bern 2003 http://www.specter.com/default50.htm
13. Provos, N. et al.: Honeyd. Ann Arbor 2004 http://www.honeyd.org
14. Shirey, R.: Internet Security Glossary. Request for comments 2828 informal. 2000
15. Spitzner, L.: Honeypots, tracking hackers. Boston 2003
16. Spitzner, L. et al.: The Honeynet Project. o. O. 2003http://www.honeynet.org
17. Stevens, R.: Kosten-, Nutzen- und Risikoanalyse für den Einsatz von Honeynets in einer Unternehmensumgebung am Beispiel des Einsatzes bei einem Internet Service Provider. Diplomarbeit Sankt Augustin 2003
18. User-Mode Linux Projekt (ed.): User-Mode Linux Kernel. o. O. 2004http://www.user-mode-linux.sourceforge.net
19. vmware Inc. (ed.): VMware Workstation/GSX Server. Palo Alto 2004