Distributed Denial of Service Angriffswerkzeuge und Abwehrmöglichkeiten

Fachgebiet Sicherheit in der Informationstechnik - TU Darmstadt
Prof. Dr. Claudia Eckert

Betreuer: Thomas Buntrock

Axel Hagedorn

Wintersemester 2002/2003

Zusammenfassung

Diese Arbeit gibt einen Überblick über die Funktionsweise und mögliche Klassifizierungen von verteilten Denial-of-Service-Attacken (DDoS) und beschreibt den prinzipiellen Aufbau von DDoS-Netzwerken zur Automatisierung dieser Attacken. Darüber hinaus werden mögliche Strategien zur Abwehr dieser Attacken untersucht.

Inhaltsverzeichnis

1 Einleitung

Verteilte Denial-of-Service (Distributed-DoS, bzw. DDoS) Angriffe stellen mittlerweile eine der größten Gefahren im und für das Internet dar. Wenn auch lange Zeit die Bedrohung durch DDoS-Attacken für von eher theoretischer Natur gehalten wurde, so mußte die Einstellung doch spätestens in den letzten drei Jahren der Einsicht weichen, mit DDoS einer ernsthaften Bedrohung gegenüber zu stehen. So zeigte erst kürzlich der Angriff auf die 13 DNS-Root-Server im Oktober 2002, welch massive Auswirkungen erfolgreiche Angriffe dieser Art in Zukunft haben könnten; vgl. dazu z.B. Dworschak (2002) und Kuri (2002/10/23). Der einstündige Angriff blieb zwar ohne größere Folgen, doch ein längerer Angriff hätte den Datenverkehr im gesamten Internet nachhaltig zu stören vermocht. Und es wäre leichtsinnig zu glauben, daß es bei diesem einen Angriff auf die Schaltzentrale des Internets bleiben wird.

Bereits im Februar 2000 machte ein größerer DDoS-Angriff von sich reden, der erstmals einen merklichen wirtschaftlichen Schaden anrichtete, indem Internetseiten unter anderem von eBay, Yahoo und Amazon lahmgelegt wurden; vgl. dazu z.B. Rötzer (2000/02/09) und Martinus (2000). Sogar das Bundesamt für Sicherheit in der Informationstechnik hat daraufhin Informationsbroschüren zum Schutz vor DDoS-Angriffen herausgegeben; vgl. BSI (2001/01/01). Die Tatsache, daß durch die freie Verfügbarkeit von Angriffswerkzeugen im Grunde jeder Internet-Nutzer zu einem potentiellen Angreifer werden kann, und daß durch die Abhängigkeit eines vernetzten Rechners von der gesamten Internet-Infrastruktur auch das am besten geschützte lokale Netzwerk zum Opfer werden kann (vgl. Houle und Weaver (2001, Oktober, S. 2)), zeigt nur die Dringlichkeit einer Auseinandersetzung mit dieser Art der Bedrohung.

Seitdem die Folgen von DDoS-Angriffen erstmals deutlich wurden, wurden viele Anstrengungen unternommen, Netze und Rechner gegen DDoS-Angriffe zu schützen - meist aber nur mit unzureichendem Erfolg. Angreifer nutzen bei ihren Attacken immer neue Sicherheitslücken in Netzen und Rechnern aus und entwickeln ihre Angriffstools, die mittlerweile Angriffe fast vollständig automatisieren, ständig weiter. Durch viele unerfahrene und vor allem im Bezug auf die Gefahren nicht sensibilisierte Nutzer des Internets mangelt es für Angriffe auch selten an den nötigen Ressourcen - die gängigsten Betriebssysteme sind voll von Sicherheitslücken bzw. in Standardinstallationen schlecht konfiguriert; vgl. Sager et al. (2000/02/28). Und selbst wenn diese Sicherheitsprobleme durch den Hersteller behoben werden, erneuern oder verändern die Anwender der Software seltenst ihre Installationen. Auf der anderen Seite werden neben dem Schließen von Sicherheitslöchern aber auch neue Strategien entwickelt, um dem Problem der DDoS-Attacken durch Absicherung der möglichen Ziele oder Übertragungswege Herr zu werden.

Diese Arbeit bietet zunächst in Kapitel 2 eine Einführung in die grundsätzliche Struktur von DoS- und DDoS-Angriffen und einen Überblick über mögliche Klassifizierungen von Angriffen. Die Beschreibung des Aufbaus eines DDoS-Netzwerks und einiger der gängigsten Werkzeuge für DDoS-Angriffe findet sich dann in Kapitel 3, bevor Kapitel 4 schließlich mögliche grundsätzliche Verteidigungsstrategien beschreibt und zwei ausgewählte Abwehrmechanismen gegen DDoS-Angriffe vorstellt. Kapitel 5 wagt einen Ausblick auf die zu erwartende Entwicklung - sowohl im Bereich der DDoS-Angriffe als auch der Maßnahmen zur Abwehr bzw. Verhinderung dieser Attacken.

2 Grundlagen

Denial-of-Service-Angriffe sind im Gegensatz zu Einbrüchen in Computersysteme oder dem Abhören von Datenströmen, die das Stehlen oder Manipulieren meist sensibler Daten zum Ziel haben, dadurch motiviert, den freien Zugang zu den vom Opfer des Angriffs angebotenen Diensten zu unterbinden. Ob es zum erklärten Ziel des Angreifers gehört oder nicht, die Folge kann ein immenser wirtschaftlicher Schaden sein - sei es dadurch, daß Kunden ausbleiben oder wichtige Informationen nicht zur Verfügung stehen.

Wie ein DoS-Angriff im Prinzip abläuft und warum gerade verteilte Angriffe so große Gefahren bergen, beschreiben die folgenden Abschnitte.

2.1 DoS - Denial of Service

Prinzipiell kann das Ziel des DoS-Angriffs auf zwei Arten erreicht werden: Entweder das System selbst, welches die betreffenden Dienste anbietet, wird geschädigt bzw. manipuliert oder der Zugriff auf die Dienste wird blockiert.

Zum ersten Fall würde z.B. das Ausnutzen einer Sicherheitslücke auf dem Zielsystem gehören, welches es einem Angreifer erlaubt, den angebotenen Dienst einfach abzuschalten, den Rechner herunterzufahren oder abstürzen zu lassen. Ganz allgemein betrachtet würde auch das Ziehen des Netzsteckers einer solchen Maschine zu dieser Kategorie der DoS-Angriffe zu zählen sein; vgl. CERT (1999/02/12). Für diese Art der Angriffe gilt jedoch, daß sie schon bei einem gesunden Maß an Vorsicht und Sorgfalt des Systemadministrators kaum noch eine Chance bekommen. (Allerdings ist gerade im Bereich der Home-User ein sehr sorgloses und problem-unbewußtes Verhalten anzutreffen, was sicher auch dazu beiträgt, daß die meisten DoS-Angriffe auf Home-Computer abzielt; vgl. Mirkovic et al. (2002b, S. 2).)

Angriffe der zweiten Kategorie nutzen für ihr Ziel den Vorteil der Masse. Sie müssen nicht unbedingt auf Sicherheitslücken des Opfers aufbauen, sondern nutzen ganz einfach die Dienste, die der Zielrechner anbietet. (Wobei natürlich auch hier Fehlkonfigurationen zum Anbieten von besonders sensiblen und angreifbaren Diensten führen können.) Der Angreifer schickt einfach so viele (meist unsinnige) Anfragen an den Zielrechner, daß dieser nicht mehr in der Lage ist, sinnvolle Anfragen anderer Nutzer zu beantworten, da er bereits für die Anfragen des Angreifers seine Rechenleistung, seinen Hauptspeicher oder die zur Verfügung stehende Bandbreite seines Netzzugangs verbraucht. Der Diensteanbieter kann bei dieser Art des Angriffs nur schwer unterscheiden, welche Anfragen zu seinem „Tagesgeschäft“ gehören und welche Teil eines Angriffs sind, was das Ergreifen von Gegenmaßnahmen sehr erschwert. Es ist allerdings leicht vorstellbar, daß bei dieser Art des Angriffs nicht nur die Ressourcen des angegriffenen Rechners aufgebraucht werden, sondern auch die des Angreifers. Eine solche Attacke kann also nur Erfolg haben, wenn der Angreifer über den besseren Rechner und die bessere Netzanbindung verfügt. Alternativ dazu kann ein solcher Angriff natürlich auch von mehreren Maschinen aus durchgeführt werden, die dann je nach Anzahl wesentlich weniger Ressourcen benötigen. In diesem Fall handelt es sich dann um einen verteilten DoS-Angriff.

Unabhängig von der Art des Angriffs schützt sich ein Angreifer vor Entdeckung durch das Fälschen seiner eigenen IP-Adresse, das sogenannte IP-Spoofing; vgl. hierzu z.B. CERT (2000/11/29) oder BSI (2002, Juli, G 5.4.2).

2.2 DDoS - Distributed Denial of Service

Um einen solchen verteilten Angriff durchzuführen, muß der Angreifer zunächst eine genügend große Anzahl von Rechnern rekrutieren, die anschließend gemeinsam den Angriff durchführen werden. Dies geschieht im allgemeinen durch das Scannen von über das Internet erreichbaren Rechnern auf Sicherheitslücken in deren Software oder Konfiguration. Gefundene verwundbare Rechner werden dann für den Angriff ausgenutzt, indem ein Progamm eingeschleust wird, welches den eigentlichen Angriff auf das Ziel der Attacke durchführt. (Natürlich kann im Zuge der Installation des Angriffsprogramms auch Code hinterlassen werden, der die Maschine auch für zukünftige Angriffe verwendbar macht.) Ein auf diese Weise durchgeführter Angriff kann eine so große Menge an Ressourcen akkumulieren, daß im Grunde kein noch so gut geschütztes Ziel in der Lage sein wird, ihm zu widerstehen und den normalen Betrieb aufrecht zu erhalten. Und selbst wenn sich das Ziel gegenüber dem Angriff behaupten kann, so ist es doch in ein lokales Netzwerk eingebunden, dessen Ressourcen ebenso in Mitleidenschaft gezogen werden. Das heißt, wenn innerhalb der Anbindung des Rechners an das Internet Router überlastet sind, oder die Netzwerkbandbreite erschöpft ist, ist das Ziel des Angriffs ebenso erreicht - selbst wenn das Opfer noch in der Lage wäre, seine Dienste anzubieten - die Anfragen erreichen den Rechner nicht mehr.

Die obigen Ausführungen lassen erkennen, daß für eine genauere Betrachtung von DDoS-Angriffen und Werkzeugen, die diese unterstützen bzw. automatisieren, eine Klassifizierung der möglichen Angriffsarten nötig ist. Im folgenden werden darum vier Kategorisierungen vorgestellt, nach denen DDoS-Angriffe unterschieden werden können; vgl. Mirkovic et al. (2002b, S. 1 - 6). Dabei werden auch Klassen von Angriffen beschrieben, die noch nicht beobachtet wurden, aber zumindest denkbar wären.

2.2.1 Automatisierungsgrad des Angriffs

DDoS-Angriffe können manuell, halb-automatisch oder automatisch durchgeführt werden. Die in der Vorbereitung eines Angriffs nötige Suche nach Rechnern, die für einen Angriff mißbraucht werden können, wurde in frühen DDoS-Attacken noch manuell ausgeführt, indem Angreifer fremde Rechner nach Sicherheitslücken absuchten, um ihre Angriffstools zu installieren und zu starten.

Mittlerweile wird diese Arbeit meist (zumindest teilweise) automatisiert: Die Rekrutierung von Agenten (Rechner, die den eigentlichen Angriff durchführen) geschieht mittels automatischer Skripte, die das Scannen nach verwundbaren Rechnern und die Installation der Angriffssoftware übernehmen. (Zu den vielfältigen Strategien, nach denen die Suche und Infizierung von Agenten ablaufen kann, sei auf Mirkovic et al. (2002b, S. 4) verwiesen.) Die Angriffssoftware muß im halb-automatischen Fall nach der Installation auf dem Agenten mit dem Handler (dem Rechner des Angreifers, der die Attacke initiiert und koordiniert) kommunizieren, um Informationen über das Angriffsziel und die Art des Angriffs zu bekommen. Dies kann entweder mittels direkter Kommunikation geschehen, bei der beide Kommunikationspartner jeweils die Adresse des anderen kennen, oder mittels indirekter Kommunikation, bei der zur Übermittlung der Anweisungen andere im Netz verfügbare Dienste zweckentfremdet werden. Die direkte Kommunikation wird dadurch ermöglicht, daß der Angriffscode bereits die Adresse des Handlers enthält. Neben dem Problem, daß sowohl Handler als auch Agenten aufgrund der offenen Verbindung durch Netzwerkscanner

aufgespürt werden können, hat diese Variante den gravierenden Nachteil, daß durch das Auffliegen eines einzelnen Agenten der gesamte Angriff vereitelt werden kann. Die indirekte Kommunikation kann z.B. auf der Basis eines IRC-Kanals etabliert werden; vgl. Houle und Weaver (2001, Oktober, S. 1). Dieser bietet allen Beteiligten genügend Anonymität und trägt dazu bei, daß die Netzwerkaktivität durch den Angriffscode unauffälliger bleibt, da sie ja einen offiziellen Service verwendet.

Bei automatischen DDoS-Angriffen wird zu Beginn das Angriffsziel, die Angriffsart und der Zeitpunkt des Angriffs bereits im Angriffscode festgelegt. Damit wird jede weitere Kommunikation zwischen den Agenten und dem Handler vermieden und der Angreifer braucht als einzige Aktion nur noch das Skript zur Verteilung des Angriffscodes starten. Diese Art des Angriffs eignet sich damit für einmalige, gezielte Aktionen.

2.2.2 Ziel des Angriffs

Bezogen auf das Ziel des Angriffs lassen sich zwei verschiedene Ansätze unterscheiden: Zum einen die Protokoll-Attacken, welche darauf abzielen, direkt bei der angegriffenen Maschine zu Ressourcen-Engpässen zu führen, und zum anderen die „Brute-Force"-Attacken, die den legitimen Zugang zum Opfer durch den Verbrauch von Netzwerkbandbreite zu unterbinden versuchen.

Protokoll-Attacken nutzen Besonderheiten oder Schwächen von bestimmten, auf einem Opfer installierten Protokollen aus. Eine der bekanntesten Attacken - das „TCP-SYN-Flooding" - nutzt die Methode der „Halboffenen Verbindungen" aus; vgl. z.B. CERT (2000/11/29). Hierbei wird vom Angreifer eine große Zahl von TCP-Verbindungsanfragen generiert, die vom Opfer angenommen werden, aber durch den Angreifer nie endgültig bestätigt werden. Der Aufbau einer TCP-Verbindung geschieht in drei Schritten: Der Client (hier der Angreifer) sendet eine SYN-Nachricht zum Server (Opfer), der den Empfang mit einer SYN-ACK-Nachricht zurück zum Client quittiert und wiederum auf eine Quittung (ACK) vom Client wartet. Im Normalfall sendet der Client diese dann auch, worauf die Datenverbindung zwischen den Rechnern genutzt werden kann. Beim SYN-Flooding enthält die SYN-Nachricht eine gefälschte IP-Adresse, womit der Server vergeblich auf eine Antwort zu seiner SYN-ACK-Nachricht wartet. Da die Anzahl der von dem Server gleichzeitig verarbeitbaren TCP-Verbindungsanfragen begrenzt ist, kann das Opfer durch eine „Überflutung" (Flooding) mit SYN-Nachrichten in einen Zustand versetzt werden, in dem es keine neuen Verbindungsanfragen mehr verarbeiten kann, da es auf die Antworten des Clients zu den halboffenen Verbindungen wartet. (Nach dem Erreichen eines Timeouts werden die dafür nötigen Ressourcen zwar wieder freigegeben, bei einer ausreichend großen Zahl von weiteren manipulierten SYN-Nachrichten aber gleich wieder belegt.) Bei fehlerhafter Software oder schlechter Administration des Servers kann dies auch den Ausfall des Rechners zur Folge haben.

Weitere Protokoll-Attacken basieren z.B. auf der Auslastung der Prozessorkapazitäten des Opfers durch das gleichzeitige Absenden von zahlreichen CGI-Anfragen oder durch das Senden von vielen falschen Signaturen an einen Authentifikationsserver.

Im Falle der Brute-Force-Attacken werden vom Angreifer sehr große Mengen von prinzipiell erlaubten und an sich ungefährlichen Paketen an das Opfer gesandt; allerdings führt die Masse der Anfragen dazu, daß das Netzwerk des Angegriffenen überlastet wird und der Rechner somit seine Dienste nicht mehr anbieten kann. Brute-Force-Angriffe bedienen sich

dabei im Grunde beliebiger Protokolle, wobei sich einige davon relativ leicht mittels einer Firewall herausfiltern lassen. Dazu gehören z.B. UDP- und ICMP-Flooding-Attacken; vgl. Houle und Weaver (2001, Oktober, S. 3).

Angriffe basierend auf Anfragen, die den vom Opfer angebotenen Diensten entsprechen, wie HTTP-Anfragen bei einem Webserver oder DNS-Anfragen an einen Nameserver, überwinden Filtermechanismen und führen neben der Überlastung des Netzwerks gleichzeitig zu einer Überlastung des Rechners selbst. Hier handelt es dann genau genommen um eine Mischung aus Protokoll- und Brute-Force-Attacke.

2.2.3 Angriffs-Dynamik

Angriffe können auch in ihrer Dynamik bzw. Dauer variieren. Neben den wohl am häufigsten anzutreffenden, kontinuierlichen Attacken, bei denen nach dem Start der Attacke alle Agenten das Ziel unter „Dauerfeuer“ nehmen, so daß ein unmittelbarer Ausfall des Ziels erreicht wird und die Attacke auch als solche zu erkennen ist, gibt es auch Attacken, die mit einer steigenden oder variierenden Paketzahl vorgehen. Mit einer langsam steigenden Anzahl von Angriffspaketen bleibt beispielsweise die Gefahr, entdeckt zu werden, weitaus geringer. Mit einer dynamischen Paketanzahl kann das Angriffsziel womöglich über längere Zeit in seiner Funktion eingeschränkt werden, ohne daß der Betreiber die Ursache für die Probleme findet.

2.2.4 Auswirkungen des Angriffs

Basierend auf den Auswirkungen des Angriffs sind zwei prinzipielle Arten zu unterscheiden: Zum einen die unterbrechende Attacke, die allein zum Ziel hat, den angebotenen Dienst des Angriffsziels vollständig außer Kraft zu setzen. Zum anderen ist aber auch eine Attacke möglich, bei der nicht der Komplettausfall des Dienstes zu erreichen versucht wird, sondern eine Einschränkung des angebotenen Service. Beispielsweise könnte gerade soviel Bandbreite verbraucht werden, daß der Rechner nur noch die Hälfte aller normalen Anfragen beantworten kann. In dieser Art der Attacke liegt ein noch viel größeres Gefahrenpotential, da sie über längere Zeit unentdeckt bleiben kann, in der aber unzufriedene Kunden abwandern oder aufgrund dessen sich das Opfer gezwungen sieht, in eine - im Grunde unnötige - Verbesserung seiner Server und seines Netzwerks zu investieren.

3 Angriffswerkzeuge für DDoS-Attacken

DDoS-Angriffe lassen sich mittlerweile mit der Hilfe von zahlreichen Software-Tools relativ leicht organisieren und durchführen. Mitte 1999 wurde der Einsatz solcher Werkzeuge mit Namen wie „Trinoo“ und „Tribe Flood Network“ (TFN) zum ersten Mal der breiten Öffentlichkeit bekannt. Sie arbeiten mittels einer Client-Server-Struktur und kombinieren mehrere bekannte Techniken zu einem komfortablen Angriffsprogramm; vgl. Strobel (2000). Seitdem wurden die Tools immer weiterentwickelt und auch kombiniert, „Stacheldraht“ z.B. vereinigt Funktionen aus Trinoo und TFN. Diese Werkzeuge ermöglichen bisher die Steuerung von bereits rekrutierten Agenten für ihren Einsatz in einer DDoS-Attacke. Die Verbreitung der Agenten-Software geschieht derzeit noch unabhängig vom Tool selbst, doch ist es wahrscheinlich nur eine Frage der Zeit, bis die Werkzeuge auch ihre eigene Verteilung übernehmen.

Die folgenden Abschnitte informieren über den prinzipiellen Aufbau der auf solchen Tools basierenden DDoS-Netzwerke und über die besonderen Eigenschaften einiger der bekanntesten Angriffswerkzeuge.

3.1 DDoS Netzwerke

Der prinzipielle Aufbau eines DDoS-Netzwerks aus Handlern und Agenten kann wie in Abbildung 1 skizziert werden; vgl. Dietrich et al. (2000, S. 330).

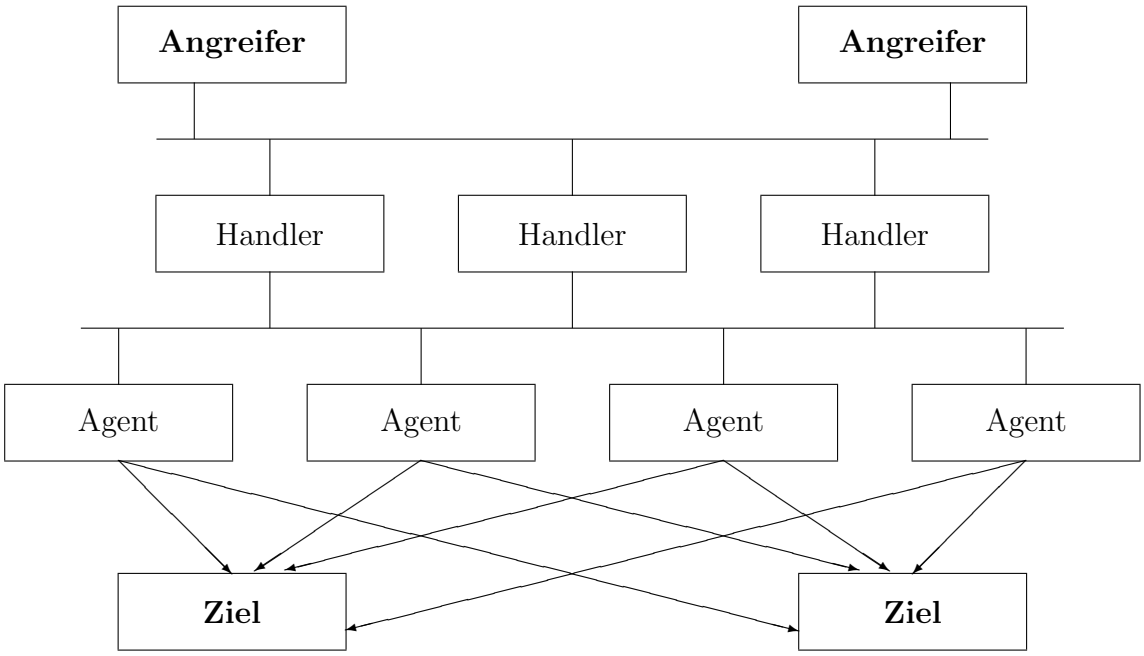

Abbildung 1: Ein typisches DDoS-Netzwerk

Der Aufbau des DDoS-Netzwerks ist hierarchisch. Ein oder mehrere Angreifer haben die Kontrolle über die Handler, mit denen wiederum die (zahlreichen) Agenten gesteuert

werden können, die letztlich ein oder mehrere Ziele angreifen. Die Kommunikation zwischen den Angreifern und den Handlern bzw. zwischen den Handlern und den Agenten, also der Steuerungs-Datenverkehr, geschieht auf der Basis von TCP, UDP, ICMP oder einer Kombination daraus. Der Angriffs-Datenverkehr zwischen Agenten und Zielen ist abhängig von der Art des (im allgemeinen Brute-Force) Angriffs. Die vier wichtigsten genutzten Angriffe sind TCP-SYN-Flooding, UDP-Flooding, ICMP-Flooding und Smurf-Attacken; vgl. hierzu z.B. CERT (2000/11/29), CERT (1997/09/24) und CERT (2000/03/13).

Zur Absicherung vor Entdeckung werden Handler nicht direkt auf dem Rechner des Angreifers gestartet, sondern auf einer zweiten Schicht und somit wie auch die Agenten auf fremden, mißbrauchten Rechnern, die in diesem Fall einen Netzzugang möglichst großer Bandbreite aufweisen und zudem möglichst viele eingerichtete Nutzer besitzen sollten, so daß auch die Chance einer Entdeckung des Handlers möglichst gering ist. Um nicht durch das Auffliegen eines Handlers das gesamte DDoS-Netzwerk zu gefährden, werden in der Praxis immer mehr als ein Handler auf verschiedenen Rechnern installiert. Darüber hinaus haben die Handler meist auch gleiche Steuerungsrechte. Der Zugriff eines Angreifers auf die Handler kann auf verschiedene Weisen erfolgen. Der einfachste Weg ist die Verwendung der Terminal-Emulation Telnet. Tools greifen jedoch meist auf durch irgendeine Art von Verschlüsselung geschützte Verbindungen zurück. Im allgemeinen verwendet ein Handler aber wiederum eine andere Methode, um mit den Agenten zu kommunizieren, damit bei Entdeckung eines Agenten nicht schon allein aufgrund der Übertragungsprotokolle auf den Angreifer geschlossen werden kann.

Der Umfang der Steuerungsmöglichkeiten (auch in Abhängigkeit von dem gewählten Kommunikationsprotokoll), die ein Handler im Bezug auf die Agenten hat, ist sehr unterschiedlich und reicht sogar bis hin zu einer Abfrage des Gesamtstatus aller Agenten.

Welche besonderen Eigenschaften einige Angriffswerkzeuge aufweisen, zeigen nun die nächsten Abschnitte.

3.2 Beispiele - Angriffswerkzeuge

Vorgestellt in ihren Grundzügen werden hier die drei klassischen Angriffstools „Trinoo“, „Tribe Flood Network“ und „Stacheldraht“. Darüber hinaus seien aber auch die Tools „Shaft“ und „Mstream“ genannt, zu denen ebenfalls umfangreiche Analysen verfügbar sind; vgl. Dietrich et al. (2000) und Dittrich et al. (2000/05/01).

3.2.1 Vorläufer

Bereits im Sommer 1998 sind erste Versuche aufgetaucht (vgl. Dietrich et al. (2000, S. 336)), die aber letztlich nur die Grundlage für die späteren Tools schufen und selbst nicht weiter in Erscheinung traten. Das erste Angriffswerkzeug „Fapi“ beherrschte schon Flooding-Attacken auf der Basis von UDP, TCP und ICMP, bot aber keine einfache Möglichkeit, ein DDoS-Netzwerk aufzubauen - zumal es auf 10 Rechner beschränkt war. Die Kommunikation zwischen Handler und Agenten erfolgte mittels UDP. Das zweite Werkzeug „Fuck_Them“ war ein reiner ICMP-Echo-Reply-Flooder, der unter anderem zufällige IP-Adressen als Absender generieren konnte.

3.2.2 Trinoo

Trinoo wurde im Frühsommer 1999 bekannt und seitdem intensiv analysiert. In einem Trinoo-Netz melden sich aktivierte Agenten (hier Daemons) am Handler (hier Master) an. Der Master seinerseits führt eine Liste aller durch ihn steuerbaren Daemons. Die Daemons sind in der Lage, sich auf dem Client-System z.B. als http-Daemon zu tarnen und somit unerkannt zu bleiben. Die Kommunikation zwischen dem Angreifer und dem Master - wie auch zwischen Master und Daemons geschieht auf der Basis von hohen UDP-Adressen. Die Daemons beherrschen als Angriff nur das UDP-Flooding.

Eine weitergehende Analyse dieses Angriffstools bietet Dittrich (1999/10/21a).

3.2.3 Tribe Flood Network

Das Tribe Flood Network (TFN) wurde ebenfalls zum ersten Mal im Frühsommer 1999 beobachtet. Das Tool soll von dem deutscher Hacker „Mixter“ stammen, der mittlerweile auch Beiträge über Angriffstools und mögliche Gegenmaßnahmen veröffentlicht hat; vgl. Dittrich (2002/12/11). Vom Aufbau her ähnelt TFN einem Trinoo-Netzwerk, unterscheidet sich aber in der Kommunikation zwischen Angreifer und Handler; dort verwendet TFN hohe TCP-Adressen. Die Kommunikation zwischen Handler und Agenten erfolgt über ICMP und wird darüber hinaus verschlüsselt. Neben den UDP-Flooding Angriffen beherrschen die Agenten auch die TCP-SYN-Attacke. In Weiterentwicklungen des Angriffstools wie Tribe Flood Network 2000 (TFN2K) wurde vor allem daran gearbeitet, die Agenten besser vor Entdeckung zu sichern.

Eine weitergehende Analyse dieses Angriffstools bietet Dittrich (1999/10/21b).

3.2.4 Stacheldraht

Das ebenfalls von einem Deutschen namens „Randomizer“ entwickelte Angriffstool Stacheldraht wurde zwar schon 1999 beobachtet, kam aber erst 2000 intensiver zum Einsatz. Stacheldraht basiert auf TFN, besitzt aber etliche Erweiterungen. So wird hier die Kommunikation zwischen Angreifer und Handler nun auch verschlüsselt und zwischen Handler und Agenten kann zusätzlich das TCP Protokoll verwendet werden. Das hat den Vorteil, daß auf dem Wirtsrechner installierte Paketfilter umgangen werden können.

Eine weitergehende Analyse dieses Angriffstools bietet Dittrich (1999/12/31).

4 Abwehrmöglichkeiten gegen DDoS-Attacken

Bisher wurden schon zahlreiche Anstrengungen unternommen, Abwehrmechanismen zu entwickeln, die Angreifer zu identifizieren und böswillige Datenströme zu stoppen versuchen. Diese mehr oder weniger erfolgreichen Abwehrversuche von DDoS-Attacken lassen sich ähnlich wie DDoS-Angriffe nach verschiedenen Merkmalen klassifizieren; vgl. Mirkovic et al. (2002b, S. 6 - 10). So ist vorerst prinzipiell zwischen präventiver und reaktiver Abwehr zu unterscheiden, während es im Bezug auf die reaktive Abwehr wiederum die beiden Teilschritte der Erkennung von Angriffen und einer Reaktion darauf zu differenzieren gilt. Bedeutend ist darüber hinaus der Grad der Kooperation von Abwehrsystemen und der Ort ihrer prinzipiellen Ansiedlung auf der Netzwerkverbindung zwischen den Angreifern und dem Opfer.

Die folgenden Abschnitte stellen auf der Basis der genannten Klassifikation Möglichkeiten zur Abwehr von DDoS-Angriffen vor, bevor schließlich zwei ausgewählte Beispiele für Abwehrsysteme beschrieben werden.

4.1 Präventive Abwehr

Eine präventive Abwehr von Angriffen kann zwei Ziele verfolgen: Die Verhinderung von Attacken (zumindest der nicht-Brute-Force-Attacken) oder die Verhinderung von DoS-Zuständen auf einem angegriffenen System.

Um das erste Ziel zu erreichen, wäre es nötig, die Systemsicherheit der zu schützenden Rechner immer auf dem aktuellsten Stand zu halten, indem verfügbare Sicherheitsupdates und Protokollfixes eingespielt und Sicherheitssoftware wie Intrusion-Detection-Systeme oder Firewalls installiert werden. Auf diese Weise können die Angriffe, die auf Sicherheitslücken im System aufbauen, unterbunden werden. Auf der anderen Seite könnte durch das gleiche Vorgehen - nur eben auf der Seite der Agenten-Rechner - die gesamte Gefahr von Brute-Force Angriffen nachhaltig eingegrenzt werden; vgl. Sager et al. (2000/02/28). (Wenn keine Systeme mehr mißbraucht werden können, sind natürlich auch keine Angriffe mehr möglich.) Eine weitere Möglichkeit stellt die grundsätzliche Verwendung von Protokollen dar, deren Kommunikationskosten nicht überwiegend oder vollständig vom Server getragen werden, sondern vom Client. Das heißt, falls zu einem Verbindungsaufbau zwischen zwei Rechnern größere Berechnungen durchzuführen sind (beispielweise bei Verschlüsselungen) sollten diese eben möglichst auf dem Client (Angreifer) und nicht auf dem Server (Opfer) ausgeführt werden. Somit wären für eine erfolgreiche Protokoll-Attacke weitaus mehr Rechner nötig als bisher.

Das zweite Ziel, die Verhinderung von DoS-Zuständen auf einem System, kann zumindest für eine ausgewählte Nutzergemeinde durch die direkte Zuteilung von Netzwerkressourcen erreicht werden. Nur die nicht bestimmten Nutzern zugeteilte Bandbreite kann dann durch Angriffe aufgebraucht werden. Eine administrativ weniger aufwendigere aber sehr kostenintensive Methode ist ganz einfach die Bereitstellung von ausreichend leistungsfähiger und redundanter Hardware, möglichst mit automatischer Lastverteilung. Diesen Ansatz verfolgen derzeit wohl die meisten größeren Firmen, unter ihnen auch Microsoft; vgl. Mirkovic et al. (2002b, S. 8).

4.2 Reaktive Abwehr

Vor einer angemessenen Reaktion auf einen Angriff steht die Erkennung desselben. Diese erfolgt mittels Mustererkennung, Anomalieerkennung oder einer Kombination aus beidem. Wenn von bekannten Angriffen Signaturen erstellt werden, können anhand dieser Muster alle passierenden Datenströme untersucht werden. Bereits bekannte Angriffe werden so sehr sicher identifiziert, jedoch vermag diese Methode keine neuen Angriffe zu erkennen. Bei der Anomalieerkennung wird der Netzverkehr mit einem „Normalmodell“ verglichen, welches den normalen Datenfluß beschreibt. Werden Abweichungen davon festgestellt, die eine gewisse Toleranzschwelle überschreiten, so können diese als mögliche Angriffe eingestuft werden. Ein offensichtlicher Nachteil hiervon ist, daß die Fehlerrate recht hoch sein kann, da die Wahl der richtigen Toleranzschwelle sehr schwierig ist. Darüber hinaus muß das Normalmodell fortwährend aktualisiert werden, um Trends abzubilden. Werden die beiden Erkennungsmethoden zu einer hybriden Methode verknüpft, können mittels der Anomalie-Erkennung automatisch neue Muster erstellt werden, die dann von der Mustererkennung weiterverwendet werden. Dieses Prinzip verwenden z.B. Intrusion-Detection-Systeme. Leider birgt es auch die Gefahr des Mißbrauchs, denn durch gezielte Manipulation könnte ein Angreifer das System dazu bringen, legitime Datenströme als Attacken zu interpretieren, sie als Muster anzulegen und somit selbst zum DoS-Tool zu werden.

Für die eigentliche Reaktion auf einen Angriff gibt es wiederum verschiedene Möglichkeiten: Es kann z.B. versucht werden, mit sogenannten Traceback-Methoden die Agenten zu identifizieren; vgl. Mirkovic et al. (2002b, S. 9). Dies allein kann den Angriff letztlich allerdings kaum verhindern, sondern trägt eher zu einer späteren Verfolgung der Täter bei. Wirksamer in der direkten Bekämpfung ist dagegen das Verringern der Bandbreite und damit das Drosseln des Datenstroms. Dies ist allerdings bei genügend starken Attacken auch nicht ausreichend, hat aber den Vorteil, daß bei unzuverlässiger Erkennung ein fälschlicherweise aggressiv eingestufter Datenstrom nicht völlig unterbrochen wird. Eine vollständige Blockade der Angriffspakete dagegen hat das Filtern zur Folge, das im Gegenzug aber wieder selbst - bei falscher Analyse des Datenstroms oder Manipulation durch einen Angreifer - als DoS-Tool fungieren kann. Eine weitere Reaktion beinhaltet eine dynamische Rekonfiguration der angegriffenen Netzwerktopologie, also zum Beispiel das Zuteilen von mehr Ressourcen oder das Isolieren einer angegriffenen Maschine.

Bei der Ausführung von Erkennung und Reaktion können die beteiligten Systeme einen unterschiedlichen Kooperationsgrad aufweisen. Neben autonomen Systemen, die lokal installiert sind, wie z.B. Firewalls, gibt es kooperierende Systeme, die ihre Erkenntnisse über Angriffe weitergeben. Beispielsweise in sogenannten Pushback-Verfahren, die das Wissen über einen Angriff an den jeweils nächsten Router weitergeben, nachdem lokale Gegenmaßnahmen ergriffen wurden (z.B. Drosselung). Somit kann der Angriff weit zurückgedrängt werden. Schließlich gibt es auch noch abhängige Systeme, die nur gemeinsam zu einem Erfolg führen. Dies liegt z.B. beim Traceback-Verfahren vor, da der Sender der Angriffspakete nur ausfindig gemacht werden kann, wenn alle Router auf dem Weg zusammenarbeiten.

Eine letzte und besonders wichtige Einteilung von Abwehrmechanismen besteht in der Wahl des Einsatzortes. Prinzipiell setzen sie an drei verschiedenen Punkten des Weges der Datenübermittlung an: Am Ziel des Angriffs, d.h. auf dem Zielrechner oder im Zielnetzwerk, im Netzwerk, d.h. auf den Routern im WAN zwischen Angreifer und Opfer, und an der Quelle des Angriffs, d.h. auf den Routern im Netzwerk des Angreifers.

Die meisten Abwehrmechanismen sind zielbasiert und können Angriffe auf das Zielsystem leicht erkennen. Sie sind jedoch nicht in der Lage, den Angriff zu stoppen, da dies die Zusammenarbeit mit den nächsten Routern erfordern würde. Systeme, die im Netzwerk zwischen Angreifer und Opfer angesiedelt sind, versuchen böswillige Datenströme zu identifizieren und ggf. zu drosseln. Dies kann einerseits automatisch geschehen oder aber auf Anforderung durch ein Opfer. Aber auch hier kann zumindest die Belastung des Netzwerks noch nicht verhindert werden. Die dritte Variante der Abwehr von DDoS-Angriffen - direkt im Netzwerk des Angreifers - ist eine Möglichkeit mit der ein Internet Dienstleister seine Kunden daran hindern könnte, DDoS-Angriffe durchzuführen. Nur stellt sich in diesem Szenario leider die Frage, wer die Kosten für das System tragen soll, da ja gerade die Nutzer des Dienstes selbst keinen Nutzen davon haben; vgl. Mirkovic et al. (2002b, S. 10).

Letztendlich besteht zusammenfassend für alle beschriebenen Kategorien von Abwehrmechanismen aber das Problem, daß bei einem entsprechend großen Angriff die Masse an Datenpaketen ganz einfach nicht bewältigt werden kann - eine hundertprozentige Lösung gibt es also (zumindest noch) nicht.

4.3 Beispiele - Abwehrmechanismen

Die folgenden in ihrer Grundidee vorgestellten Beispiele zeigen zwei sehr vielversprechende Ansätze, deren Umsetzung aber eher fraglich und zumindest in nächster Zeit wohl nicht realisierbar ist. Weitere Beispiele für Abwehrsysteme finden sich u.a. bei Dittrich (2002/12/11).

4.3.1 D-WARD

D-WARD (DDoS Network Attack Recognition and Defense) ist ein sehr aktueller Ansatz zur Abwehr von DDoS-Angriffen direkt an der Quelle des Angriffs, entwickelt von Mirkovic et al. (2002a). Es setzt „Policies“ für die Adressen in dem betreuten Netzwerk, die festlegen, welche Nutzer was tun dürfen. Des weiteren setzt es auf Normalmodelle für den Netzwerkverkehr - insbesondere prüft es, ob auf vom lokalen Netz ausgesandte Datenpakete auch ausreichend korrekte Antworten zurückkommen; beispielsweise die ACK-Meldungen bei einem TCP-Verbindungsaufbau. Werden nicht modellkonforme Datenströme ermittelt, so werden sie intensitätsabhängig in der ihnen zur Verfügung stehenden Bandbreite beschnitten. Allein hätte solch ein System allerdings nur Erfolg, wenn es flächendeckend eingesetzt würde, was sicher nicht zu erreichen ist.

4.3.2 AEGIS

Das von Chen (2001) vorgeschlagene AEGIS-System überträgt die Idee der mobilen Agenten auf Router. Mittels auf Routern residierenden aktivem Code (eigenständig agierend), der sich selbst propagieren kann, wird ein aktives Netzwerk etabliert. Als Voraussetzung müssen sich alle im AEGIS-Netz existierenden aktiven Router kennen. Erkennt der aktive Code auf einem Router einen Angriff, so propagiert er alles Wissen über diesen Angriff an alle ihm bekannten Router im Netz, wodurch erreicht wird, daß Gegenmaßnahmen auf allen Routern im Netz vorgenommen werden können und der Angriff so vom eigentlichen Ziel ferngehalten wird. Es existieren zwar Testumgebungen für AEGIS, doch für einen Einsatz dieses Systems fehlt derzeit noch die Hardware, die aktive Komponenten unterstützt.

5 Ausblick

Aufgrund der steigenden Bedeutung des Internets, vor allem im Bereich des eCommerce, wird die Auseinandersetzung mit dem Problem DDoS immer wichtiger. Die rasend schnelle Evolution von DDoS-Werkzeugen in den letzten drei Jahren läßt bezüglich der Häufigkeit und Intensität von zukünftigen Angriffen kaum eine Abnahme vermuten. Vor allem ist auch damit zu rechnen, daß solche Tools immer schwieriger zu entdecken sein und so die Bekämpfung weiter erschweren werden.

Zum aktuellen Zeitpunkt kann im Grunde nur versucht werden, durch Aufklärung beim Benutzer und durch Druck auf die Hersteller ein größeres Problembewußtsein beim normalen Computer-Anwender für Systemsicherheit zu erzeugen, bzw. Betriebssysteme in ihrer Standardkonfiguration ohne freigeschaltete Dienste auszuliefern. Grundsätzlich kann zwar jeder an das Internet angeschlossene Rechner für eine DDoS-Attacke mißbraucht werden, vor allem aber sind es UNIX/Linux basierende Systeme, die am häufigsten für Angriffe genutzt werden; vgl. CERT (2000/03/03). Hier sind neben den vielen Linux Anfängern vor allem Universitäten angesprochen, die nicht nur die leistungsfähigsten Rechenzentren betreiben sondern auch den einfachsten Zugang zu den Ressourcen auf ihren UNIX-Systemen ermöglichen. Erst wenn durch entsprechende Maßnahmen die Möglichkeit zum Mißbrauch von Rechnern als Angriffs-Agent einer DDoS-Attacke genommen wird, wird die Gefahr durch solche Angriffe eingeschränkt werden können. Bis dahin bleibt derzeit als einzig wirksames Mittel nur die redundante Installation von gefährdeten Computersystemen und der Ausbau der Bandbreite zu diesen Systemen.

Wie sich Abwehrmechanismen und einzelne Lösungen weiterentwickeln werden, bleibt fraglich, denn meist scheitern die bisherigen Lösungen daran, zu weitreichende Veränderungen an der Internet-Infrastruktur vornehmen zu müssen. Es fehlt definitiv noch das einfach zu installierende und lokal einsetzbare Abwehrtool. Allerdings lassen sich auch im Hardwaremarkt Entwicklungen ausmachen, die in Zukunft davon ausgehen lassen können, daß das Aufspüren von Angriffsrechnern und evtl. auch die Abwehr einfacher wird; vgl. CISCO (2002). Weiterhin werden grundsätzliche Änderungen im Aufbau des Internets und der darin verwendeten Protokolle Auswirkungen auf die Ausführbarkeit von DDoS-Angriffen haben. Beispielhaft seien hier die Einführung von IPv6 oder IPsec genannt, die IP-Spoofing enorm erschweren werden.

Für eine weitere Beobachtung der Entwicklungen im Bezug auf die Analyse und Abwehr von DDoS-Attacken sei neben den aktuellen Mitteilungen des CERT (2002) die Linksammlung von Dittrich (2002/12/11) empfohlen.

Literatur

BSI (2001/01/01): *Maßnahmenkatalog gegen DDoS-Angriffe - Kurzinformationen zu aktuellen Themen der IT-Sicherheit*. Bundesamt für Sicherheit in der Informationstechnik.
URL http://www.bsi.bund.de/literat/faltbl/faltddos.htm. 1

BSI (2002, Juli): *IT-Grundschutzhandbuch - Standard-Sicherheitsmaßnahmen*. Bundesamt für Sicherheit in der Informationstechnik.
URL http://www.bsi.de/gshb/deutsch/menue.htm. 2.1

CERT (1997/09/24): *UDP Port Denial-of-Service Attack*. CERT Coordination Center, Advisory CA-1996-01.
URL http://www.cert.org/advisories/CA-1996-01.html. 3.1

CERT (1999/02/12): *Denial of Service Attacks*. CERT Coordination Center.
URL http://www.cert.org/tech_tips/denial_of_service.html. 2.1

CERT (2000/03/03): *Denial-of-Service Tools*. CERT Coordination Center, Advisory CA-1999-17.
URL http://www.cert.org/advisories/CA-99-17-denial-of-service-tools.html. 5

CERT (2000/03/13): *Smurf IP Denial-of-Service Attacks*. CERT Coordination Center, Advisory CA-1998-01.
URL http://www.cert.org/advisories/CA-1998-01.html. 3.1

CERT (2000/11/29): *TCP SYN Flooding and IP Spoofing Attacks*. CERT Coordination Center, Advisory CA-1996-21.
URL http://www.cert.org/advisories/CA-1996-21.html. 2.1, 2.2.2, 3.1

CERT (2002): *CERT Homepage*. CERT Coordination Center.
URL http://www.cert.org/. 5

Chen, E. Y. (2001): *AEGIS: An Active-Network-Powered Defense Mechanism against DDoS Attacks*. Active Networks, IFIP-TC6 Third International Working Conference, IWAN 2001, Philadelphia, PA, USA, 30. September - 2. Oktober 2001, **LNCS 2207**, S. 1 – 15. 4.3.2

CISCO (2002): *IP Source Tracker*.
URL http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s21/ipst.pdf. 5

Dietrich, S.; N. Long und D. Dittrich (2000): *Analyzing Distributed Denial of Service Attack Tools: The Shaft Case*. Proceedings of the 14th System Administration Conference, New Orleans, LA, USA, 3. - 8. Dezember 2000, **LISA 2000**, S. 329 – 339.
URL http://www.adelphi.edu/~spock/lisa2000-shaft.pdf. 3.1, 3.2, 3.2.1

Dittrich, D. (1999/10/21a): *The DoS Project's „trinoo" distributed denial of service attack tool*.
URL http://staff.washington.edu/dittrich/misc/trinoo.analysis.txt. 3.2.2

Dittrich, D. (1999/10/21b): *The „Tribe Flood Network“ distributed denial of service attack tool*.
URL http://staff.washington.edu/dittrich/misc/tfn.analysis.txt. 3.2.3

Dittrich, D. (1999/12/31): *The „Stacheldraht“ distributed denial of service attack tool*.
URL http://staff.washington.edu/dittrich/misc/stacheldraht.analysis.txt. 3.2.4

Dittrich, D. (2002/12/11): *Distributed Denial of Service (DDoS) Attacks/tools*.
URL http://staff.washington.edu/dittrich/misc/ddos/. 3.2.3, 4.3, 5

Dittrich, D.; G. Weaver; S. Dietrich und N. Long (2000/05/01): *The „mstream“ distributed denial of service attack tool*.
URL http://staff.washington.edu/dittrich/misc/mstream.analysis.txt. 3.2

Dworschak, M. (2002): *Angriff der Geisterarmee*. Der Spiegel **55** (44), S. 96. 1

Houle, K. J. und G. M. Weaver (2001, Oktober): *Trends in Denial of Service Attack Technology*. CERT Coordination Center.
URL http://www.cert.org/archive/pdf/DoS_trends.pdf. 1, 2.2.1, 2.2.2

Kuri, J. (2002/10/23): *Denial-of-Service-Attacke gegen DNS-Rootserver*. Heise NewsTicker.
URL http://www.heise.de/newsticker/data/jk-23.10.02-001/. 1

Martinus, K. (2000): *Dienstblockade - DDoS-Attacken auf internationale Websites*. iX **12** (4), S. 97 ff. 1

Mirkovic, J.; J. Martin und P. Reiher (2002a): *Attacking DDoS at the Source*. Proceedings of the 10th International Conference on Network Protocols, Paris, France, November 2002, **ICNP 2002**, S. 312 – 321.
URL http://lasr.cs.ucla.edu/ddos/404_mirkovic_j.pdf. 4.3.1

Mirkovic, J.; J. Martin und P. Reiher (2002b): *A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms*. Technischer Bericht 020018, University of California, Computer Science Department, Los Angeles. 2.1, 2.2, 2.2.1, 4, 4.1, 4.2

Rötzer, F. (2000/02/09): *ECommerce-Websites lahmgelegt*. Telepolis.
URL http://www.heise.de/tp/deutsch/inhalt/te/5766/1.html. 1

Sager, I.; N. Gross und J. Carey (2000/02/28): *Locking Out the Hackers - How to safeguard the Web*. Business Week Online.
URL http://www.heise.de/newsticker/data/jk-23.10.02-001/. 1, 4.1

Strobel, S. (2000): *Untergründiges - Distributed-Denial-of-Service-Angriffe*. iX **12** (8), S. 102 ff. 3