DDoS-Angriffs- und Verteidigungsstrategien

Alexander Wittmann
Betreuer: Marc Fouquet
Seminar Innovative Internet Technologien und Mobilkommunikation SS2010
Lehrstuhl Netzarchitekturen und Netzdienste
Fakultät für Informatik, Technische Universität München
Email: alexander.wittmann@in.tum.de

KURZFASSUNG

Distributed Denial-of-Service (DDoS) Angriffe stellen eine aktuelle Bedrohung für Netzwerkumgebungen dar. Die Vielzahl und Vielfalt sowohl der Angriffs- als auch der Verteidigungsansätze ist beachtlich.

Dieses Dokument präsentiert und erläutert die wichtigsten und bedeutendsten Möglichkeiten zum Angriff und zur Verteidigung, und bietet damit Neulingen die Möglichkeit zum besseren Verständnis des Problems und der aktuellen Lösungsansätze.

Der die Angriffsmöglichkeiten betreffende Teil der Ausarbeitung stellt Gemeinsamkeiten heraus und hebt wichtige Merkmale der Strategien hervor. Der die Verteidigungsstrategien betreffende Teil erläutert die Vor- und Nachteile der Lösungsansätze.

Schlüsselworte

DoS, DDoS, Attack Rate, Source Address Validity, Spoofing, Traceback, Pushback, Source Address Filtering, Overlay Filtering, Anomaly Detection, Client Puzzles

1. EINLEITUNG

Dieses Dokument gibt einen allgemeinen Überblick über Angriffe, deren primäres Ziel es ist, den Zugriff auf eine bestimmte Ressource zu verweigern. Im speziellen werden einige Möglichkeiten vorgestellt, wie auf solche Angriffe reagiert werden kann.

Als Denial of Service (kurz DoS, englisch für: Dienstverweigerung oder -ablehnung) wird in der digitalen Datenverarbeitung die Folge einer Überlastung von Infrastruktursystemen bezeichnet (siehe Abbildung 1).

Solch eine Dienstverweigerung kann durch unbeabsichtigte Überlastungen verursacht werden oder durch einen mutwilligen Angriff auf einen Host (Server), einen Rechner oder sonstige Netzkomponenten in einem Datennetz. Dies geschieht in der Regel mit der Absicht, einen oder mehrere bereitgestellte Dienste arbeitsunfähig zu machen.

Erfolgt solch ein Angriff koordiniert von einer größeren Anzahl anderer Systeme aus, so spricht man von Verteilter Dienstblockade oder englisch Distributed Denial of Service (DDoS). Normalerweise werden solche Angriffe nicht per Hand, sondern mit Backdoor-Programmen oder Ähnlichem durchgeführt, die sich von alleine auf anderen Rechnern im Netzwerk verbreiten und dem Angreifer durch solche Botnetze weitere Wirte zum Ausführen seiner Angriffe bringen.[4]

Nicht alle Dienst-Ausfälle, auch diejenigen, die sich aus bösartigen Aktivitäten ergeben, sind zwangsläufig Denial-of-Service-Attacken.

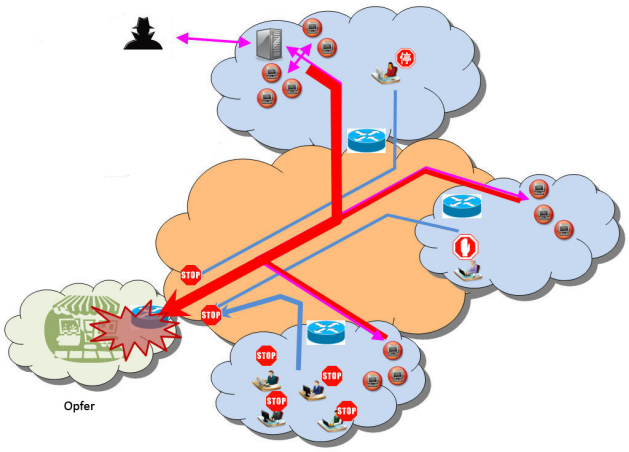

Abbildung 1: DDoS-Angriff

Unrechtmäßige Nutzung von Ressourcen kann auch zu Dienst-Ausfällen führen. Zum Beispiel kann ein Eindringling einen anonymen FTP-Bereich als Ort nutzen um illegale Kopien von kommerzieller Software zu speichern, somit Speicherplatz verbrauchen und die Generierung von Netzwerk-Verkehr verursachen. Derartige Angriffe sollen aber nicht im Rahmen dieses Dokuments behandelt werden.

Denial-of-Service-Angriffe können Rechner und Rechnernetze wesentlich beeinträchtigen. Je nach Art Ihres Unternehmens, kann somit gezielt Ihr Geschäftsbetrieb zum Erliegen gebracht werden. So kam es bereits häufiger vor, dass Anbieter von E-Commerce Diensten erpresst wurden. Sollte einer Lösegeldforderung nicht nachgekommen werden hätte dies eine Blockade der Internetpräsenz durch einen DDoS-Angriff zur Folge. Der entstehende Schaden ist immens.

Einige Denial-of-Service-Angriffe können mit einfachen Mitteln gegen einen leistungsfähigen Host eingesetzt werden. Diese Art von Angriff wird als ein asymmetrischer Angriff bezeichnet. So ist es z.B. einem Angreifer möglich mit ge-

ringem Rechenaufwand und geringer Bandbreite erheblich leistungsfähigere Endgeräte zu beeinträchtigen.

2. ANGRIFFSSTRATEGIEN

Denial-of-Service-Attacken gibt es in einer Vielzahl von Formen. Die Angriffe zielen dabei auf unterschiedliche Dienste ab. Es gibt einige grundlegende Arten von Angriffen. Man kann dabei zwei grundsätzliche Typen unterscheiden:

- Verbrauch knapper, beschränkter oder nicht-erneuerbaren Systemressourcen
- Überflutung des Netzwerks mit Angriffen um legitimen Verbindungsaufbau zu verhindern

2.1 Ausschöpfen von Systemressourcen

Teilnehmer eines Netzwerkes benötigen bestimmte Ressourcen um ihre Dienste anbieten zu können, wie etwa Speicher- und Festplattenplatz, CPU-Zeit, und Datenstrukturen.

Denial-of-Service-Attacken richten sich am häufigsten gegen Netzwerkerreichbarkeit. Das Ziel ist es, Hosts oder Netzwerke an der Kommunikation zu hindern. Ein Beispiel für diese Art von Angriff ist der SYN-Flood-Angriff, der im Folgenden kurz beschrieben wird:

Bei dieser Art von Angriff, baut der Angreifer eine Verbindung zum Rechner des Opfers auf, ohne aber den Verbindungsaufbau zu vollenden. In der Zwischenzeit hat das Opfer eine bestimmte Anzahl an Systemressourcen reserviert, um die bevorstehende vollständige Verbindung vorzubereiten.

Das Ergebnis ist, dass legitime Verbindungen verweigert werden, während die Opfer-Maschine darauf wartet, dass halboffene Verbindungen vollständig aufgebaut werden.

Es ist zu beachten, dass diese Art von Angriff nicht davon abhängt, dass der Angreifer in der Lage ist die vorhandene Netzwerkbandbreite vollständig aufzubrauchen. In diesem Fall verbraucht der Angreifer Ressourcen, welche am zur Verfügung stellen der Netzwerkverbindung beteiligt sind.

Das bedeutet also, dass ein Angreifer diesen Angriff von einer langsamen Verbindung aus gegen einen leistungsfähigen Server in einem sehr schnellen Netzwerk ausführen kann. (Dies ist ein gutes Beispiel für einen asymmetrischen Angriff.)

2.2 Ausschöpfen von Netzwerkbandbreite

Ein Angreifer kann auch in der Lage sein, die komplette zur Verfügung stehende Bandbreite in einem Netzwerk aufzubrauchen, indem er eine große Anzahl von Paketen an das Netzwerk sendet. Typischerweise sind diese Pakete ICMP ECHO-Pakete, unterliegen aber prinzipiell keiner Einschränkung.

Darüber hinaus muss der Angreifer nicht von einer einzigen Maschine aus operieren. Er kann in der Lage sein den gemeinsamen Angriff mehrerer Maschinen zu koordinieren, um einen erheblich stärkeren Effekt zu erzielen. (Dies ist ein gutes Beispiel für einen DDoS-Angriff aus einem Bot-Netz)

3. ANGRIFFSEIGENSCHAFTEN

Bei beiden zuvor genannten Typen lassen sich folgende Angriffseigenschaften unterscheiden:

- Angriffe mit echten oder mit gefälschten Absender IP-Adressen
- Angriffe mit dynamischem Wirkungsgrad, die nicht dazu dienen das Netzwerk unverzüglich lahm zu legen sondern auf Dauer die Bandbreite zu begrenzen ohne erkannt zu werden

3.1 Authentizität der Absenderadresse

Source-IP-Spoofing - also das Fälschen der Absender IP - spielt eine wichtige Rolle bei DDoS-Attacken. Würde es eliminiert werden, könnten viele Arten von DDoS-Attacken durch Ressourcenmanagement-Techniken gelöst werden, was eine gerechte Verteilung der Host- oder Netzwerkressourcen, an jede Quell-IP-Adresse bedeuten würde. Basierend auf der Gültigkeit der Absenderadresse, unterscheidet man zwischen Angriffen mit gefälschten Absenderadressen und gültigen Absenderadressen.

DDoS-Angriffe mit gefälschter IP-Absenderadresse sind die vorherrschende Art von Angriffen, da es ein Vorteil für den Angreifer ist, z.B. um die Zurechenbarkeit zu vermeiden, und um durch die größere Adressvielfalt die Erkennung des Angriffs zu erschweren.

Angreifer können nicht routbare Source-Adressen vortäuschen, von denen einige auf eine Reihe von reservierten Adressen (zB 192.168.0.0/16) zeigen oder Teil eines zugeteilten, aber nicht verwendeten Adressraums eines Netzwerks sind. Angriffspakete, die reservierte Adressen verwenden, können leicht erkannt und verworfen werden, während die Pakete die keine reservierten Adressen benutzen wesentlich schwieriger zu entdecken sind.

Viele Angriffe fälschen zufällige Absenderadressen in den Angriffspaketen, da dies einfach durch die Generierung von zufälligen 32-Bit-Zahlen, mit denen die Pakete versehen werden, erreicht werden kann.

Beim sogenannten Subnetz-Spoofing fälscht ein Angreifer eine zufällige Adresse aus dem zugewiesenen Adressbereich in dem sich die Maschine befindet. Zum Beispiel könnte eine Maschine, die Teil des Netzwerks 131.179.192.0/24 ist, jede beliebige Adresse im Bereich 131.179.192.0 - 131.179.192.255 fälschen. Da Maschinen in einem Subnetz sich das Medium (Ethernet) teilen um die Gateway-Router (First Hop auf dem Weg zur Außenwelt) erreichen zu können, kann Spoofing nur von diesem Router nachgewiesen werden. Es ist unmöglich, die Fälschung irgendwo außerhalb des Gateway Routers zu erkennen.

Angreifer profitieren vom Quell-Adress-Spoofing und sind dazu geneigt, es einzusetzen, wann immer es geht. Es ist jedoch nicht mehr dringend notwendig auf Spoofing zu setzen. Zum einen kann ein großer Teil der Angriffe erkannt werden (siehe später), zum anderen hat das Aufkommen von Bot-Netzen dazu beigetragen, das ein Initiator eines DDoS-Angriffs sich einer Vielzahl von Hosts mit unterschiedlichen Adressen bedienen kann.

So können Angriffe von sehr vielen unterschiedlichen Hosts aus durchgeführt werden. Die hohe Zahl der Angreifer macht es möglich, dass das Paketaufkommen eines einzelnen am Angriff beteiligten Rechners sehr gering sein kann und somit die Wahrscheinlichkeit sinkt, dass dessen singuläre Aktivität als böswillig eingestuft wird. Durch die hohe Zahl der Angreifer kann das Ziel dennoch leicht zum erliegen gebracht werden.[3]

3.2 Dynamik der Angriffsrate

Während des Angriffs sendet jeder der teilnehmenden Angreifer einen Strom von Paketen an das Opfer. Ausgehend vom Mechanismus, welcher die Rate ändert, unterscheiden wir zwischen Angriffen mit konstanter, steigender und schwankender Intensität.

Die meisten bekannten Angriffe gehen mit konstanter Rate vor. Nach dem Beginn des Angriffs erzeugen die Angreifer Pakete in gleichmäßigem Tempo, in der Regel so viele wie ihre Mittel es erlauben. Die plötzliche Flut an Paketen stört die Dienste des Opfers schnell. Dieser Ansatz bietet das beste Kosten-Nutzen-Verhältnis für den Angreifer, da er eine minimale Anzahl von Clients bereitstellen muss, um Schaden zuzufügen. Auf der anderen Seite kann der große, kontinuierliche Verkehrsstrom als Anomalie erkannt werden und erleichtert so die Angriffsentdeckung.

Angriffe mit variabler Paketrate verändern die Intensität des Paketstroms der angreifenden Maschine um die Aufdeckung des Angriffs zu verzögern oder zu vermeiden.

Angriffe, die einen stufenweisen Anstieg der Rate verwenden, führen zu einem langsamen Erschöpfen der Ressourcen des Opfers. Die zur Verfügung gestellten Dienste des Opfers werden über einen langen Zeitraum hinweg immer etwas mehr beeinträchtigt. Somit kommt es zu einer wesentlich verzögerten Erkennung des Angriffs.

Der Schaden dieses Angriffs kann sehr groß sein, da zwar nur ein geringer Teil der Ressourcen beeinträchtigt wird, dies allerdings über einen langen Zeitraum hinweg. Ein Angriff mit hoher Rate und dem Zweck, das Netzwerk zum erliegen zu bringen würde schneller erkannt und beseitigt werden.

Angriffe die mit fluktuierender Rate arbeiten passen die Angriffsrate, ausgehend von dem Verhalten des Opfers oder beruhend auf einem vorprogrammierten Timing an, um der Linderung des Schadens und der Entdeckung zu entgehen.

Bei einem pulsierenden Angriff brechen die Angreifer in wiederkehrenden Abständen die Attacke ab, um sie später fortzusetzen. Wenn dieses Verhalten gleichzeitig von allen Angreifern durchgeführt wird, erfahren die Opfer regelmäßige Service-Unterbrechungen. Wenn die Angreifer jedoch in Gruppen unterteilt sind, die so koordiniert werden, dass eine Gruppe immer aktiv ist, dann erfährt das Opfer kontinuierliche Dienstverweigerung, während es die anhaltende Anomalie gar nicht als solche erkennen kann.[3]

4. VERTEIDIGUNGSSTRATEGIEN

Internet Denial-of-Service (DoS) Attacken fluten begrenzte Ressourcen mit Anfragen und verhindern damit legitimen Nutzern den Zugriff auf diese Ressource. Zu den Zielen gehören die Bandbreite an Netzzugangspunkten und anderen Netzwerk-Engpässen, und auch die Rechen- und Speicher-Ressourcen auf Servern, Clients, Routern und Firewalls. Zum Beispiel werden einige Low-End-Router lahmgelegt, wenn an sie Pings in einer zu hohen Rate gesendet werden, weil die CPU überfordert ist.

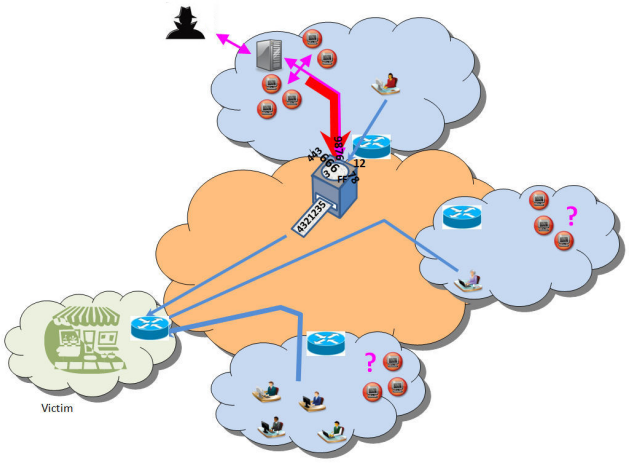

Abbildung 2: Source Address Filtering

Es mag zwar nahe liegen, dieses Problem durch den Einsatz robusterer Software zu lösen. Dennoch ist jedes mit dem Internet verbundene Gerät verwundbar durch einen Flooding-Angriff.

Da DoS-Attacken immer häufiger auftreten wurden viele Wissenschaftler zu Forschungsarbeiten motiviert. Diese Arbeit soll neben gängigsten Angriffsmethoden auch wichtige Erkennungs- und Abwehrmöglichkeiten behandeln.

DoS-Lösungen werden meist anhand ihrer Realisierbarkeit und ihrer Wirksamkeit bei der Abwehr bewertet. Während dies zweifellos richtig ist, soll die Aufmerksamkeit auch auf eine andere Art von Problem auf diesem Gebiet gelenkt werden: die künftigen Folgen für das Internet und seine Anwendungen, falls diese Ansätze zum Einsatz kommen sollten. Ein guter Schutz vor DoS-Lösung muss nicht nur wirksam sein, er muss es auch erlauben neue Netzdienste nahtlos einzuführen. Vor dem Hintergrund aktueller politischer Entwicklungen muss auch beachtet werden, dass die Methoden zur DDoS-Abwehr auch sehr leicht zur Internetzensur eingesetzt werden könnten.

4.1 Reaktive Methoden

4.1.1 Source-Address-Filtering

Einer der frühesten Vorschläge um DoS-Angriffe abzuschwächen war es, Source-Adress-Filtering an allen Netzwerk Einstiegs- und Ausstiegspunkten bereitzustellen (siehe Abbildung 2). Dies würde Angreifer daran hindern beliebige Absenderadressen in ihren Paketen anzugeben und wäre somit nützlich um die Anzahl an Arten der Angriffe zur reduzieren. Ein Source-Filter lässt sich als Filter beschreiben, der Pakete abweist, die einen Punkt im Netzwerk nicht auf legitime Art und Weise erreicht haben können.

Filtering ist nur dann effektiv, wenn es in großem Maße ein-

gesetzt wird. Eine Quelladresse kann den Nachweis für die Urheberschaft nur dann erbringen, wenn jeder Knoten im Netzwerk einen Filter implementiert um die Abgeschlossenheit des Systems sicherzustellen. Obwohl es in den letzten Jahren als best practice galt, gibt es noch viele Lücken bei der Abdeckung mit Adressfiltern. Selbst wenn vollständige Abdeckung gegeben wäre - Fortschritte bei den Angriffsmethoden haben die Relevanz der Filter zur Bedeutungslosigkeit verkommen lassen.

Hinter einem Filter können alle Source-Adressen, die innerhalb eines Netzwerkpräfixes liegen, gefälscht werden. Dies können tausende von Adressen sein. Hinzu kommt, dass automatisierte Botnetz-Tools es leicht gemacht haben, eine sehr große Anzahl an Hosts für einen bestimmten Angriff zu gewinnen. Angreifer gehen heute oft mit legitimen, ungefälschten Pakete vor. Wenn eine Million unterschiedlicher Rechner im Netz ein einziges TCP-SYN-Paket senden, spielt es keine Rolle, dass sie alle ihre reale Absenderadresse verwenden.

4.1.2 Traceback and Pushback

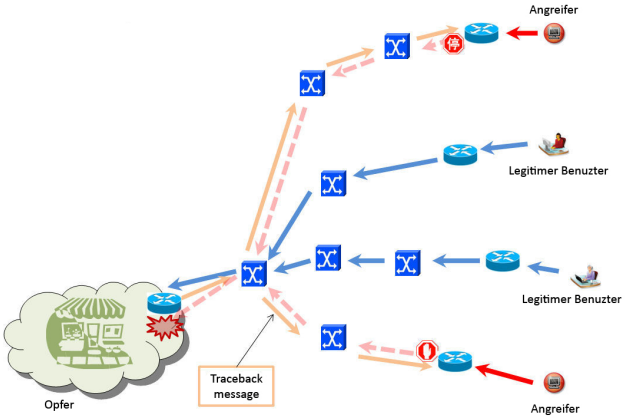

Abbildung 3: Traceback

Es wurden mehrere Methoden einen Angriff zur Quelle zurück zu verfolgen entwickelt. Traceback konzentriert sich auf die Identifizierung der Hosts, die für einen Angriff verantwortlich sind, unternimmt aber ebenso wie Source-Filterung wenig, um den Angreifer am Senden von Datenverkehr zu hindern (siehe Abbildung 3). DoS-Angriffe werden in der Regel von einer (möglicherweise sehr großen) Zahl an kompromittierten Rechnern aus gestartet. Traceback Mechanismen können bei der Ermittlung solcher kompromittierter Hosts von unschätzbarem Wert sein. Dabei erstellt das angegriffene Ziel eine Signatur der Pakete, die den Angriff verursachen. Diese Signatur wird allen dem Ziel vorgelagerten Routern mitgeteilt. So können diese überprüfen, woher der böswillige Datenstrom kommt, diesen zurückverfolgen und selbst die Signatur an benachbarte Router weiterleiten. So kann der Ursprung des Angriffs ermittelt werden. Das aber ist zu spät und den Angriff zu verhindern, oder den initiierenden Täter des selbigen zu ermitteln (wenn der Angriff von einem Bot-Netz ausgeht).

Negativ fällt auf, dass die Implementierung des Systems Performanceeinbrüche verursacht. Außerdem ist man auf die Unterstützung der Internet Service Provider angewiesen, da das Filter-System erst auf den im Internet verbreiteten Router installiert werden muss.

Um diese Einschränkung zu umgehen wurden die beteiligten Traceback-Router mit Filterfunktionalitäten ausgestattet. Mit Pushback (siehe Abbildung 4) kennzeichnet ein Knoten, welche Art von Paketen den Angriff verursacht, und sendet Anfragen an vorgelagerte Knoten deren Paketrate bereits näher an der Quelle zu verringern. Obwohl sich die heutigen Pushback-Vorschläge darauf konzentrieren die Bandbreite einer Verbindung während eines Flooding-Angriffs zu kontrollieren, könnte theoretisch jeder beliebige Host im Internet dynamisches Pushback verwenden, um die Erschöpfung seiner Ressourcen zu verhindern.

Leider kann es schwierig sein Filter zu entwerfen, die perfekt zwischen gewolltem und ungewolltem Traffic unterscheiden. Einstufung anhand der Paket-Header ist anfällig für Spoofing. Einstufung anhand des Paket-Inhaltes wird vereitelt durch die zunehmende Verwendung von Ende-zu-Ende Verschlüsselung. Zudem wird durch die Integration von Filtern eine Grundlage für Zensur geschaffen.

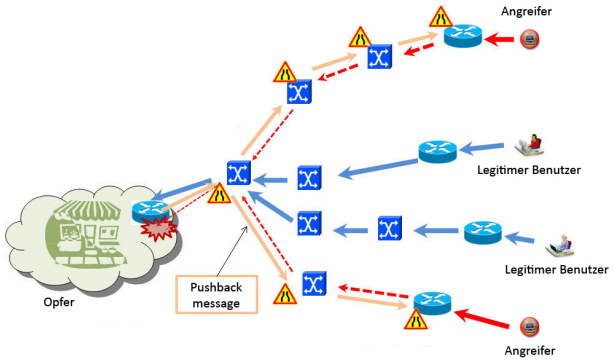

Abbildung 4: Pushback

Ausgeklügelte Angriffe können auch Sonden einsetzen um die Filterfunktion herauszufinden und sie dann zu umgehen. Beispielsweise begrenzt die erste Pushback-Implementierung einfach die Rate aller Pakete mit demselben Ziel. Dieser Ansatz wird nicht nur ebenso gut böswillige Pakete blockieren, er kann auch leicht durch einen Angriff umgangen werden bei denen Pakete mit unterschiedlichem Ziel über denselben Knoten geroutet werden, welcher dann den Flaschenhals darstellt. [5]

4.1.3 Overlay Filtering

Overlays (siehe Abbildung 5) wurden angesichts der hohen Dauer um anspruchsvolle Filter auf der Router-Hardware hinzuzufügen als ein Konzept vorgeschlagen, das es ermöglicht DoS-Filtering Schrittweise zu implementieren. Hier wird der komplette Traffic an ein angegriffenes Ziel über spezielle Zwischenknoten geroutet. Da diese nicht auf dem normalen Routingpfad liegen, können besondere Analyse- und Filtermethoden angewandt werden.

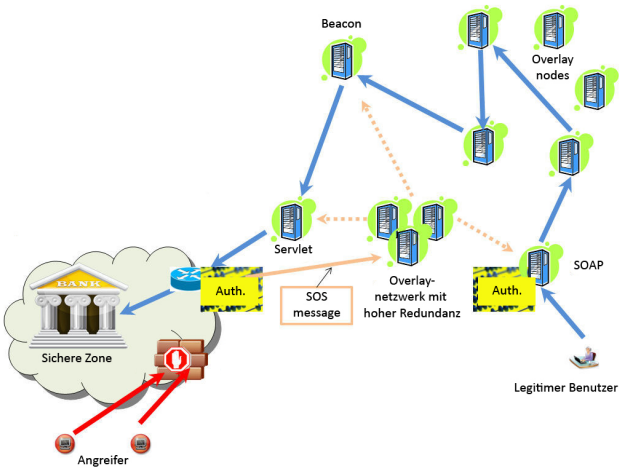

Abbildung 5: Overlay Filtering

Das zu erreichende Ziel (z.B. eine sicherheitskritische Website) wird dabei über einen geheimen Proxy an das Netzwerk angebunden, dem es Dienste zur Verfügung stellen soll. Der Weg zu diesem Proxy führt über ein großes Netz an Overlay-Knoten (Redundanz, falls Pfade im Netz wegen z.B. DDoS nicht erreichbar sind). Der Nutzer der das Overlay-Netzwerk verwenden will, muss sich an einem Einstiegspunkt authentifizieren. Von dort wird er dann an die Overlay-Knoten weitergeleitet. Diese finden den Weg zum Proxy anhand einer geheimen Information im Protokollheader. Router die auf dem Weg zum Ziel liegen können dann so konfiguriert werden, dass alle Pakete, die keine geheime Information beinhalten verworfen werden.

Overlay Filtering hat den Vorteil, dass der Zugriff auf das Netz nur nach stattgefundener Authentifizierung möglich ist. Allerdings sind Angriffe auf die Einstiegspunkte möglich. Wird der Proxy bekannt sind auch dort Angriffe möglich.[1]

4.1.4 Anomaly Detection

Der vielleicht aktivste Bereich in der DoS-Prävention ist Anomalie-Erkennung. Regelbasierte oder statistische Verfahren werden verwendet, um Traffic-Muster in freundlich oder böswillig einzuteilen. Nur eine automatische Reaktion auf einen DoS-Angriff oder eine Sicherheitsverletzung kann schnell genug sein, um Beschädigungen oder Verlust des Dienstes zu verhindern.

Es bestehen allerdings große Bedenken über die Folgen der Implementierung solcher Systeme. Letztlich ist Anomalie-Erkennung keine ausreichende Antwort auf das Problem - die Entscheidung, ob ein bestimmter Datenfluss einen Angriff darstellt oder nicht, muss an den Verbindungsendpunkten auf Anwendungsebene erfolgen.

Schlimmer noch, führt die Anomalie-Erkennung zu abgeschlossenen Systemen, denn ISPs und Systemadministratoren werden sämtliche Traffic-Formen, die nicht standardisiert sind blockieren, um den Wettlauf mit dem Angreifer zu gewinnen. Da Filter-Policies in der Regel geheim sind werden viele Entwickler legitimer Anwendungen nie genau wissen, warum der Verkehr einen Alarm ausgelöst hat beziehungsweise zurückgewiesen wurde.

Wie kann z.B. eine Applikation wissen, dass ein Low-End-Router Ping-Pakete nur bis zu einer bestimmten Rate akzeptiert bevor eine Überlastsituation eintritt, wenn der Router keine Möglichkeit hat, den Absender über seine Ressourcen-Limits zu informieren?

Die Einführung neuer Netzdienste verkäme praktisch zu einer Unmöglichkeit: eine neue Anwendung müsste ihr Datenaufkommen außerordentlich konservativ bemessen, um eine unverhältnismäßige Reaktion von Filter-Systemen oder Netzwerkadministratoren zu vermeiden.[2]

4.2 Präventive Methoden

4.2.1 Client-Puzzles

Client-Puzzles (siehe Abbildung 6) sind eine in der Entwicklung befindliche, viel versprechende Technik, die Service-Garantie für den rechtmäßigen Nutzer ermöglichen will. Diese erhalten den Zugang zu einer Dienstleistung erst, nachdem sie ihre Legitimität nachgewiesen haben. Für jede Service-Anfrage, ist der Nutzer gezwungen, ein kryptographisches Rätsel zu lösen, bevor der Server seine Ressourcen freigibt. Dies stellt eine große Aufgabe für den Angreifer dar, wenn er Traffic in großen Mengen generieren will.

Die Grundidee hinter Client-Puzzles ist, dass jeder Client der den Dienst des Servers ersucht, einige seiner eigenen Ressourcen (Rechenzeit oder Speicher, etc.) verwenden muss, bevor der Server seine Ressourcen für die Verbindung zugesteht. Dies schützt vor Angriffen, die von einer großen Anzahl von Botnetz Computern inszeniert werden, welche echte IP-Adressen verwenden, da die bestehenden DDoS-Tools so sorgfältig konstruiert sind, dass sie den Zombie-Computer nicht zu stören versuchen, um den Eigentümer nicht auf ihre Anwesenheit aufmerksam zu machen.

Dieser Ansatz zur DoS Prävention scheint interessant zu sein, nicht nur aufgrund der formalen Modellierung, sondern auch aufgrund des Protokoll-Designs. Es ist nicht erforderlich, dass Angriffssignaturen und verdächtiger Datenverkehr durch die beteiligten Router und Filter erkannt werden. Da es sich um einen präventiven Ansatz handelt, wird kein legitimer Datenverkehr unterbunden werden. Dies ist auf die Verwendung von Client-Puzzles zurückzuführen. Die Fähigkeit, Rätsel zu lösen, trennt berechtigte Nutzer von automatischen Angriffs-Tools.

Client-Puzzles sind ein Mechanismus, um präventiv gegen DoS-Angriffe vorzugehen. Es ist den Clients möglich, für den Dienst des Servers zu bieten. Die geschieht durch die Berechnung von Rätseln mit unterschiedlichen Schwierigkeitsgraden. Der Server passt den Schwierigkeitsgrad abhängig von der aktuellen Auslastung an. Bei einer Anfrage kann der Client diese entweder akzeptieren, oder er sendet eine Absage an den Server.

Die Clients können dann ihre Leistung erhöhen um den Schwierigkeitsgrad des Rätsels zu bewältigen und senden die Anfrage an den Server zurück. Legitimierte Clients bekommen somit Zugriff auf den Server durch die Erhöhung des Schwierigkeitsgrades, während ein Angreifer weniger Anreiz hat das zu tun, denn er würde nicht wollen, dass der Besitzer des

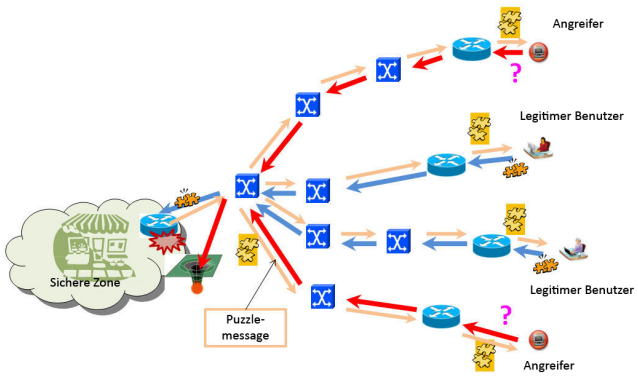

Abbildung 6: Client-Puzzles

befallenen Rechners durch übermäßigen Rechenaufwand benachrichtigt wird.

Ein Client Puzzle könnte etwa ein Java-Script in einer Website sein, dass aus einem gegebenen Hashwert per Brute-Force Methode ein verschlüsseltes Secret berechnen sollte. Über den Grad der Verschlüsselung lässt sich der Schwierigkeitsgrad bestimmen.

Standard Client-Puzzle-Protokolle funktionieren gut um das Aufbrauchen von Ressourcen auf dem Server bei Angriffen zu verhindern. Sie scheitern jedoch, wenn der Angreifer direkt eine riesige Welle an Paketen an einen bestimmten Server sendet, um dessen Bandbreite zu erschöpfen. Um diesen bösartigen Verkehr zu blockieren, muss ein Filter-Mechanismus in den dazwischen liegenden Netzwerken verwendet werden, anstatt auf Server-Ebene.

5. ZUSAMMENFASSUNG

Wie nun gezeigt wurde, besteht ein ständiger Wettkampf zwischen organisierten Kriminellen beim Finden neuer Angriffsmethoden und den Entwicklern geeigneter Gegenmaßnahmen. Diese Maßnahmen sind leider auch immer mit der Einschränkung des Komforts des Benutzers bzw. mit Einbußen bei der Einführung und Administration von Netzdiensten verbunden.

Um auf die vielfältigen Angriffsmöglichkeiten wie Ressourcen- und Bandbreitenausschöpfung, SYN-Floods, ICMP-Floods, IP-Spoofing und Botnetze reagieren zu können genügt es nicht einen einzigen Lösungsansatz zu verfolgen.

Es stehen mit den gegebenen Filter-Methoden, mit Traceback- und Pushback-Ansätzen, mit Overlay-Netzwerken einige hilfreiche Werkzeuge zur Verfügung, um Angriffe abzuwenden. Besonders der Bereich Anomalieerkennung erfährt zur Zeit große Beachtung in der Wissenschaft.

Vielversprechend sind die neuen, sich in der Entwicklung befindlichen spieletheoretischen Ansätze, bei denen zwischen Client und Server eine Nutzenfunktion optimal erfüllt werden muss, damit Zugriff gewährt wird oder nicht. Client-Puzzles stellen im Moment eine aussichtsreiche Lösung dar, die das Resource-Exhaustion Problem beseitigen könnte.

Leider kann keine der aufgeführten Lösungen alle Herausforderungen lösen. Für ISPs und Administratoren gilt es daher, eine geeignete Kombination der vorhandenen Werkzeuge einzusetzen. Die Kaskadierung der einzelnen Lösungen scheint angebracht. So sollte grundlegende Sicherheit durch hoch verbreitete Filter-Methoden schon in den ISP-Netzen gegeben werden. Auch muss die Unterstützung für Traceback und Pushback Möglichkeiten sowie Anomalieerkennung gegeben sein. Um den Benutzer so wenig wie möglich zu beeinträchtigen sollten weitere Methoden wie z.B. Client-Puzzles allerdings erst (und dies natürlich automatisiert) zum Einsatz kommen, wenn ein akuter Angriff vorliegt.

6. LITERATUR

[1] A. D. Keromytis, V. Misra, and D. Rubenstein. Sos: secure overlay services. In *SIGCOMM '02: Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 61–72, New York, NY, USA, 2002. ACM.

[2] A. Mahimkar and V. Shmatikov. Game-based analysis of denial-of-service prevention protocols. In *CSFW '05: Proceedings of the 18th IEEE workshop on Computer Security Foundations*, pages 287–301, Washington, DC, USA, 2005. IEEE Computer Society.

[3] J. Mirkovic and P. Reiher. A taxonomy of ddos attack and ddos defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34:39–53, 2004.

[4] Wikipedia. Denial of service. Website, 2010. Available online at `http://de.wikipedia.org/w/index.php?title=Denial_of_Service&oldid=75812106` visited on June, 25th of 2010.

[5] J. Xu and W. Lee. Sustaining availability of web services under distributed denial of service attacks. *IEEE Trans. Comput.*, 52(2):195–208, 2003.