Detektion und Prävention von Denial-of-Service Amplification Attacken – Schutz des Netzes aus Sicht eines Amplifiers

Timm Böttger[1], Lothar Braun[1], Oliver Gasser[1],
Helmut Reiser[2], Felix von Eye[2]

[1] Technische Universität München (TUM), 85748 Garching b. München

[2] Leibniz-Rechenzentrum (LRZ), 85748 Garching b. München

[boettget,braun,gasser]@net.in.tum.de

[reiser,voneye]@lrz.de

Zusammenfassung

Distributed Denial-of-Service Angriffe versuchen, durch Überlastsituationen Dienste oder Systeme zu sabotieren. Ein spezieller Angriffstyp ist dabei der Amplification Angriff, bei dem sich legitime Systeme und Dienste – die sogenannten Amplifier – unwissentlich an den Angriffen beteiligen, indem sie auf gespoofte Anfragen antworten. Die Besonderheit dabei ist, dass die Antworten auf diese Anfragen von der Größe der Pakete oder auch von der Anzahl der Pakete größer sind als die Anfrage.

Um den Amplifier bzw. das Netz des Amplifiers zu schützen, wurde ein Ansatz entwickelt, der in der Lage ist, mit hoher Wahrscheinlichkeit zu erkennen, ob ein Amplification Angriff durchgeführt wird. Diese Information kann dann verwendet werden, um den Angriff entweder zu unterbinden oder um ihn weiter zu analysieren.

Der Ansatz wurde erfolgreich in einem größeren Netz eines Hochschulrechenzentrums getestet und konnte dort die gestellten Anforderungen übertreffen: Zum einen konnten schon bekannte und beobachtete Angriffe erfolgreich detektiert werden, zum anderen war es möglich, neue potentielle Angriffsmethoden zu erkennen.

Schlussendlich wird eine Verwertungsmöglichkeit der Erkennungsergebnisse gegeben, indem aufgezeigt wird, wie die Prävention von Amplification Attacken realisiert werden kann.

1 Einleitung

Klassische netzbasierte Angriffe kennen zwei verschiedene Rollen. Auf der einen Seite ist der Angreifer, der mit seinem Angriff ein bestimmtes Ziel verfolgt, auf der anderen Seite

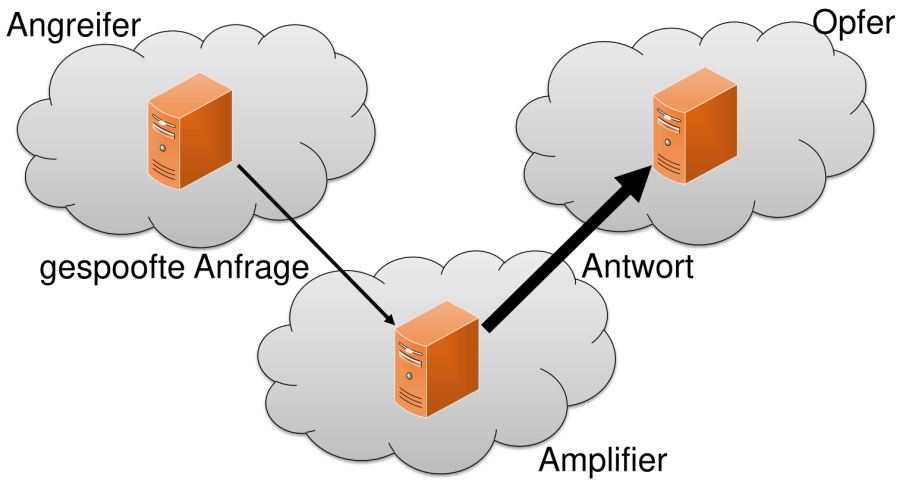

Abbildung 1: Vereinfachtes Modell einer Amplification Attacke

das Opfer, das Ziel des Angriffs ist. Dabei ist es zumeist unerheblich, wie viele Angreifer oder Opfer bei einem speziellen Angriff beteiligt sind. Klassische Schutzmaßnahmen wie Firewalls, Intrusion Detection Systeme oder auch Programme wie fail2ban[1] sind darauf ausgerichtet, Angriffsversuche auf der Opferseite möglichst frühzeitig zu erkennen bzw. zu verhindern, um den potentiellen Schaden so gering wie möglich zu halten.

Bei Amplification Attacken wird dieses Rollenmodell um eine weitere Komponente vergrößert. Diese weitere Rolle, der sogenannte Amplifier, ist weder im klassischen Sinne vom Angreifer kompromittiert noch anderweitig unter seiner Kontrolle, sondern bietet im Allgemeinen einen regulären Dienst an. Dabei machen sich die Angreifer aber eine Besonderheit dieser regulären Dienste zunutze: Sie senden ein relativ kleines Paket als Anfrage an den Amplifier und dieser antwortet darauf mit einem deutlich größeren Antwortpaket. Werden nun viele Anfragepakete von einer gespooften Adresse aus gesendet, verstärkt der Amplifier diesen Angriff je nach verwendeten Dienst deutlich und wird somit Teil einer größer angelegten Denial-of-Service-Attacke, die zum Ziel hat, Systeme und Dienste des Opfers zu stören, wie auch in Abbildung 1 vereinfacht dargestellt ist.

Dass Amplification Attacken durchaus beträchtliche Datenraten produzieren können, vor allem wenn durch eine kombinierte und verteilte Aktion mehrere Amplifier beteiligt sind, zeigen einige Berichte [Sol14], die von Datenraten von mehr als 100 GBit/s sprechen oder der in [Pri13] berichtete Angriff auf die Anti-Spam-Blockliste Spamhaus, die im März 2013 durch eine konzentrierte Aktion mit Datenraten von bis zu 300 GBit/s umgehen musste.

In der bisherigen Forschung [Ros14] haben sich 14 UDP-basierte Protokolle als angreifbar gegenüber Amplification Attacken herausgestellt. Dabei variiert der Amplification Faktor, also der Faktor zwischen Anfrage- und Antwortgröße, sehr stark zwischen den Protokollen, auch ist die Verbreitung der einzelnen Dienste, die diese Protokolle verwenden, sehr unterschiedlich. So sind die verwundbaren DNS- oder NTP-Dienste recht weit verbreitet und somit flächendeckend gefährdet, während Quake3-Installationen – mit Ausnahme von einschlägigen Gaming-Providern – nicht sehr häufig anzutreffen sind.

[1] `http://www.fail2ban.org`

Die Erkennung von Amplification Attacken kann an mehreren Stellen durchgeführt werden, wobei zwei der offensichtlichen Erkennungspunkte, beim Opfer oder auch beim Angreifer, in dieser Arbeit keine Rolle spielen sollen, da der Fokus nur auf dem Netz des Amplifiers liegen soll. Dabei liegt die Schwierigkeit darin, dass weder das Spoofing sicher erkannt werden kann, noch dass sich die reinen Anfragen eines Amplification Angriffs ohne weiteres von legitimen Anfragen unterscheiden.

Gerade im Betrieb eines Rechenzentrums oder eines daran angeschlossenen Netzes ist es jedoch wichtig zu erkennen, wenn eigene Systeme durch einen Angreifer für Amplification Angriffe ausgenutzt werden, da Amplification Angriffe zum einen einen erhöhten Supportaufwand in der Behebung des Problems und dem Bearbeitung der externen Beschwerden verursachen, zum anderen aber auch durch die missbräuchliche Nutzung der eigenen Ressourcen gegebenenfalls Engpässe entstehen können. Dabei ist insbesondere von Interesse, einen allgemein gültigen und somit generischen Ansatz zu haben, mit dessen Hilfe auch bisher unbekannte und unerforschte Angriffe erkannt werden können.

Um dem Anspruch, eine generische Erkennung zu gewährleisten, die unabhängig von dem verwendeten Protokoll arbeitet, erfüllen zu können, wird in diesem Paper eine neuartige Methode vorgestellt, wie die Erkennung von Amplification Angriffen auf UDP-basierte Protokolle innerhalb des Netzes des Amplifiers realisiert werden kann. Diese Methode ermöglicht die Charakterisierung von UDP-Verkehr anhand einiger ausgewählter Kriterien. Diese Methode wird im Kapitel 3 präsentiert, nachdem im Kapitel 2 der aktuelle Stand der Forschung und verwandte Arbeiten im Bereich Amplification Angriffe und deren Detektion besprochen werden.

Um die Ergebnisse der Arbeit evaluieren zu können, wurde die Infrastruktur und das daran angeschlossene Netz eines größeren Hochschulrechenzentrums verwendet. Dieses wurde zum einen gezielt angegriffen, um zu testen, ob die Erkennungsmechanismen greifen, zum anderen wurde der normale Nutzungsverkehr mit in die Untersuchungen mit eingebunden. Diese Evaluierung wird im Detail in Kapitel 4 beschrieben.

Neben der Detektion von Angriffen spielt ebenso die Prävention von Angriffen eine große Rolle, da dadurch die Ressourcen, die für eine Detektion und Behebung von Störungen und Angriffen eingesetzt werden müssen, sinnvoller verwendet werden können. Durch die Informationen, die während der Detektion gesammelt werden, kann die Prävention in diesem Bereich sinnvoll verbessert werden. Dabei spielt eine große Rolle, dass es zwar recht einfach ist, offene UDP-Ports zu erkennen, die dahinter laufenden Dienste zu ermitteln jedoch sehr zeit- und ressourcenintensiv ist. Somit ist es sinnvoll, die Erkennung auf die Protokolle und Dienste zu beschränken, bei denen es wahrscheinlich ist, dass sie ausgenutzt werden können, so wie dies im hier vorliegenden Fall der Amplification Angriffe der Fall ist. Kapitel 5 beschreibt näher, wie die Erkennung von Amplification Angriffen genutzt werden können, zukünftige Amplification Angriffe präventiv zu verhindern.

Schlussendlich fasst Kapitel 6 die Arbeit noch einmal zusammen und gibt einen Ausblick auf zukünftige Arbeiten.

2 Verwandte Arbeiten

Denial-of-Service Angriffe existieren bereits seit einer langen Zeit. In der Vergangenheit wurden unterschiedliche Arten von DoS-Angriffen beschrieben [SL02]. Amplification Angriffe sind eine Unterart der verfügbaren DoS Angriffstechniken. Einige von ihnen, wie der Smurf-Angriff [CER98] mit Hilfe des ICMP-Protokolls, sind schon seit mindestens 1998 bekannt.

In den letzten Jahren verschob sich der Fokus von Ampflication Angriffen jedoch mehr zu UDP-basierten Protokollen. Speziell solche UDP-Anwendungen, die keinen gemeinsamen geteilten Zustand zwischen Client und Server benötigen, können potentiell ausgenutzt werden. Über lange Zeit hinweg war DNS ein wichtiges Beispiel für UDP-basierte Amplification Angriffe. Daher wurden speziell zur Erkennung solcher Angriffe verschiedene Methoden vorgestellt, mit denen DNS-basierte Ampflication-Angriffe erkannt werden können [KMGG07b, SLS08, KMGG07a, RSR09].

Im Februar 2014 stellte Christian Rossow eine neue Studie vor, in der er 14 UDP-basierte Protokolle identifizierte, die verwundbar für Ampflication Angriffe sind [Ros14]. In seiner Arbeit beschreibt er, wie diese Angriffe durchgeführt werden können, und analysiert die möglichen Auswirkungen solcher Angriffe. Von einigen der vorgestellten Protokolle war bekannt, dass sie ausnutzbar waren. Angriffe auf andere Protokolle wurden jedoch zum ersten Mal beschrieben. Es ist jedoch bekannt, dass weitere Protokolle verwundbar sind: So stellte z.B. Özavci einen Amplification Angriff vor, der das SIP Protokoll ausnutzt [Fat13]. Basierend auf diesen neuesten Erkenntnissen, nehmen wir an, dass es möglicherweise weitere ausnutzbare Protokolle gibt, die bisher noch nicht bekannt sind.

In seiner Arbeit stellt Rossow auch zwei Algorithmen vor, mit deren Hilfe man Ampfliciation Angriffe im Netz des Opfers und des Amplifiers erkennen kann. Er definiert dazu den sogenannten Bandwidth Amplification Factor (BAF), der die Menge des erzeugten Verkehrs zwischen Client und Server beschreibt. Sein Algorithmus erkennt als einen Alarm, wenn ein zu großes Missverhältnis zwischen der Menge des Verkehrs, die ein Client versendet und der Menge des Verkehrs, die ein Server versendet besteht. Rossow beschränkt die Analyse von Netzwerkverkehr auf die Ports der verwundbaren Protokolle, die einen fixen Port verwenden. Er beschränkt sich damit auf eine Untermenge der verwundbaren Protokolle und kann somit zum Beispiel keinen Ampflication Angriff mittels des BitTorrent-Protokolls erkennen. Weiterhin können damit auch keine Angriffe auf Protokolle erkannt werden, von denen bisher noch nicht bekannt ist, dass sie verwundbar sind.

In dieser Arbeit zeigen wir, dass der Ansatz von Rossow eine hohe Menge von False-Positives erzeugen kann, wenn man ihn portunabhängig eingesetzt und den gesamten Netzwerkverkehr untersucht. Weiterhin stellen wir einen neuartigen Ansatz vor, der mit hoher Wahrscheinlichkeit erkennen kann, ob ein Ampflication Angriff auf einem beliebigen Port mittels eines beliebigen UDP-basierten Protokolls durchgeführt wird.

3 Detektion von Amplification Angriffen

Um Amplification Attacken erkennen zu können, muss zunächst die Kommunikationsbeziehung zwischen Client und Server betrachtet und modelliert werden. Das klassische Konzept

der Netzwerkflows ist in diesem Fall nicht anwendbar, da manche Protokolle (bspw. DNS) für jede Anfrage einen neuen UDP-Port verwenden, so dass eine eine Kommunikationsbeziehung in einer Vielzahl von Flows resultieren würde. Zur Beschreibung von Kommunikationsbeziehungen verwenden wir daher einen sog. *Pairflow* für jedes Server- / Clientpaar:

$$pairflow := < C_{IP}, S_{IP}, S_{port}, B_{2s}, B_{2c}, t > .$$

In dem Sechs-Tupel des Pairflows ist C_{IP} die IP des Clients, S_{IP} und S_{port} die IP und der Port des Servers. Im Falle einer Amplification Attacke ist der Server in der Rolle des Amplifiers. Da der Angreifer die IP-Adresse des Opfers spooft, verbirgt sich hier aus Netzsicht hinter der Client IP sowohl der Angreifer als auch das Opfer. Mit Hilfe eines Pairflows wird also jegliche Kommunikation zwischen einem Client und einem bestimmten Serverdienst beschrieben. Weiterhin beinhaltet der Pairflow Informationen darüber wie viele Byte in Richtung des Servers (B_{2s}) und in Richtung des Clients (B_{2c}) gesendet wurden sowie einen Zeitstempel (t). Diese Modellierung entspricht der Modellierung, die auch schon Rossow in [Ros14] vorgeschlagen hat.

Ein Amplification Angriff zeichnet sich dadurch aus, dass der Amplifier signifikant mehr Daten in Richtung des Opfers schickt als andersherum. Diese Eigenschaft lässt sich mit Hilfe des *Bandwith Amplification Factors (BAF)* beschreiben. Formal ist der BAF definiert als:

$$BAF = \frac{len(UDP\ payload)\ server\ to\ client}{len(UDP\ payload)\ client\ to\ server}.$$

Wir erwarten, dass Amplification Angriffe einen Bandwith Amplification Factor oberhalb eines gewissen Schwellwertes aufweisen werden. In dieser Arbeit setzen wir den Schwellwert auf fünf, erwarten also, dass im Falle eines Angriffes der Amplifier mindestens fünfmal mehr Verkehr in Richtung des Opfers schickt als er von diesem erhält. Besonders Pairflows, die nur wenig Verkehr widerspiegeln, können leicht große BAF-Werte erreichen. Zur weiteren Filterung setzen wir daher voraus, dass der Amplifier mindestens 10 MB Verkehr in Richtung des Opfers erzeugt. Dieser Wert wurde während der in Kapitel 4 beschriebenen Evaluation für das dort benutzte Hochschulrechenzentrum ermittelt und kann für andere Institutionen auch einen anderen Wert annehmen. Setzt man aber diesen Schwellenwert zu klein an, so steigt die Gefahr für False-Positiv-Meldungen. Als Aktiv- und Inaktivtimeout verwenden wir jeweils zehn Minuten. Dieser Filterprozess ist deckungsgleich zur Filterung, die von Rossow et. al in [Ros14] vorgenommen wird.

Alle Amplification Angriffe sollten die bislang beschriebenen Kriterien erfüllen, da mit ihrer Hilfe der Verstärkungsteil präzise gefasst wird. Allerdings gibt es auch viele legitime Anwendungsfälle, in denen Verkehrsmuster erzeugt werden, die obige Kriterien erfüllen. Beispielhaft seien hier große Downloads, Dateiaustausch über Peer-to-Peer-Netzwerke oder VPN-Verbindungen genannt. Um solchen Verkehr nicht fälschlicherweise als Angriff zu deklarieren, wird die Erkennung in [Ros14] auf die UDP-Ports solcher Serverdienste beschränkt, von denen eine Verwundbarkeit gegenüber Amplification Angriffen bekannt ist. Um eine zuverlässige Erkennung zu gewährleisten, ohne sich auf bestimmte Serverdienste bzw. Ports zu beschränken, werden dementsprechend weitere Unterscheidungsmerkmale benötigt.

3.1 Eigenschaften von Amplification Angriffen

Um weitere Kriterien zur Erkennung von Amplification Attacken zu bestimmen, ist es sinnvoll, sich in einen potentiellen Angreifer hineinzuversetzen.

Ein Angreifer sendet Anfragen an den Amplifier in der Erwartung, dass der Amplifier auf diese mit einer größeren Antwort reagiert. Da der Angreifer hierfür die IP-Adresse des Opfers spoofen muss, wird er im Allgemeinen die Antworten des Amplifiers nicht sehen können. Es ist dem Angreifer also weder möglich, einen gemeinsamen Zustand (shared state) mit dem Amplifier aufzubauen, noch kann er sich sicher sein, dass seine Anfragen die gewünschte Reaktion des Amplifiers hervorrufen. Er kann daher keine beliebigen Anfragen stellen, sondern nur solche, die sich beantworten lassen, ohne einen Zustand vorhalten zu müssen. Ein potentieller Angreifer wird weiterhin versuchen, mit möglichst geringem Aufwand und Risiko einen größtmöglichen Schaden zu verursachen. Zum Ausführen einer Amplification Attacke wird der Angreifer zunächst solche Anfragen identifizieren, auf die der Amplifier mit möglichst großen Antworten reagieren wird. Für den nachfolgenden Angriff wird der Angreifer nur die Anfrage (bzw. die wenigen Anfragen) verwenden, die den größten Verstärkungsfaktor aufweisen, da der Angreifer an einem möglichst großen Schaden interessiert ist. Aufgrund dieser Einschränkungen der möglichen Anfragen erwarten wir, dass der Angreifer während einer Amplification Attacke viele sehr ähnliche Nachrichten verschicken wird. Da die Antworten des Amplifiers zustandslos sind, erwarten wir auch hier, dass die Antworten einander sehr ähnlich sein werden.

Gelingt es dem Angreifer, dass der Amplifier Antworten an das Opfer schickt, so wird das Opfer viele unerwartete Nachrichten (unsolicited messages) erhalten, da das Opfer diese Antworten nicht angefordert hat. Sofern die Ressourcen des Opfers noch nicht vollkommen erschöpft sind, sollte es auf diese unerwarteten Nachrichten mit ICMP port unreachable Nachrichten reagieren.

Ausgehend von diesen Vorüberlegungen haben wir Kriterien identifiziert, die wir zur weiteren Differenzierung von legitimen und Angriffsverkehr verwenden können.

3.2 Kriterien zur Angriffserkennung

Ähnlichkeit der Anfragen und Antworten: Wir erwarten, dass der Angreifer Nachrichten ähnlicher Größe versendet, da er einen großen Amplification Faktor erreichen möchte und daher möglichst kleine Nachrichten verschicken wird. Da der Angreifer nur eine limitierte Anzahl an unterschiedlichen Anfragen verwendet, versendet auch der Amplifier nur eine geringe Anzahl an unterschiedlichen Antworten. Daher erwarten wir auch hier, dass die Antwortgrößen des Amplifiers über verschiedene Antworten hinweg sehr ähnlich sein werden. Wir erfassen daher die Paketgrößen getrennt für jede Richtung der Kommunikation.

Weiterhin erwarten wir, dass die Payloads der Anfragen und Antworten untereinander ähnlich sein werden. Wiederum deshalb, da der Angreifer nur wenige verschiedene Anfragen verwendet und der Amplifier aufgrund er zustandslosen Arbeitsweise nur mit einer begrenzten Anzahl an Antworten reagieren kann. Um die Ähnlichkeit der Payloads zu bestimmen, komprimieren wir die UDP-Payloads der Pakete mit dem Deflate-Algorithmus [Deu96]. Um

Rechenleistung zu sparen, speichern wir für jeden Pairflow, der die BAF-Kriterien erfüllt, die Payloads der nächstne 100 Pakete in jede Richtung und wenden die Kompression nur auf diese an. Wir konkatenieren die Payloads und berechnen dann für jede Kommunikationsrichtung einen Kompressionsfaktor (Compression Factor (CF)) als

$$CF = \frac{len(deflate(concatenated\ UDP\ payload))}{len(concatenated\ UDP\ payload)}.$$

Ein CF-Wert nahe 0 bedeutet, dass die Payloads gut komprimierbar und daher sehr ähnlich sind. Ein CF-Wert nahe 1 impliziert, dass die Payloads nicht seht gut komprimierbar und damit recht unterschiedlich waren.

Unerwartete Nachrichten: Die Nachrichten des Amplifiers an das Opfer sind unerwartet, der Netzwerk Stack das Opfers sollte also mit ICMP unreachable Nachrichten reagieren, wenn UDP-Nachrichten empfangen werden, die an einen Port gerichtet sind, auf dem keine Anwendung läuft. Wir erfassen daher die Anzahl der ICMP unreachable Nachrichten, die das Opfer an den Amplifier schickt. Hier könnte man zwar davon ausgehen, dass eine oder mehrere ICMP unreachable Nachrichten ausreichen sollten, um zu signalisieren, dass die Gegenseite die verschickten Antworten nicht empfangen will oder kann. Jedoch muss man sich immer bewusst machen, dass ICMP Nachrichten im Allgemeinen gerade beim domänenübergreifenden Austausch von Nachrichten nicht immer zuverlässig sind. Weiterhin ist insbesondere bei mobilen Geräten nicht sichergestellt, dass die Netzanbindung zu jeder Zeit ausreichend stabil ist, so dass unter Umständen die Antworten des Servers genau zu dem Zeitpunkt beim Nutzer eintreffen, wenn dieser keine Netzanbindung mehr hat. Eine dann verschickte ICMP unreachable Nachricht würde in dem Fall nur signalisieren, dass der Nutzer (kurzfristig) nicht erreichbar ist, aber nicht, dass er die Anfrage nicht gestellt hat.

IP Spoofing: Ein wesentlicher Teil des Amplification Angriffs ist das IP Spoofing. Gelingt es, Indizien zu sammeln, die für IP-Spoofing sprechen, so ist auch dies ein Indikator, dass ein Angriff vorliegt. Es ist möglich, aus dem IP-Header der eingehenden Pakete aus dem TTL-Feld die Wegstrecke bis zum Sender zu schätzen. Die initialen TTL-Werte werden vom Betriebssystem gesetzt und können sich je nach Betriebssystem unterscheiden[2]. Jeder Router auf dem Weg dekrementiert das TTL-Feld um eins. Wir erfassen die TTL der eingehenden Pakete und berechnen die Pfadlänge, indem die beobachtete TTL von der nächst-größeren bekannten Initial-TTL eines Betriebssystems abgezogen wird. Wenn wir von dem Opfer ICMP-Replies sehen können, extrahieren wir auch hier die TTL und bestimmen die Pfadlänge. Treten beim Vergleichen dieser beiden Pfadlängen Unstimmigkeiten auf, so kann dies ein Hinweis auf IP-Spoofing sein. Beobachten wir keine ICMP-Replies des Opfers, so führen wir zur Bestimmung der Pfadlänge zwischen Server und Client aktive Traceroutes durch.

Jedoch kann die Ermittlung der Pfadlänge kein alleiniges Kriterium sein, um festzulegen, ob es sich bei der aktuellen Verbindung um einen IP-Spoofing-Versuch handelt, da insbesondere bei asymmetrischen Routing oder bei größeren Distanzen die verwendeten Pfade von IP-Paket zu IP-Paket unterscheiden können. Somit bleibt dieses Kriterium schließlich nur ein Indiz.

[2]Linux und viele BSD-Varianten verwenden eine initiale TTL von 64, der Netzwerkstack von Windows verwendet 128 und manche Unix Varianten verwenden eine initiale TTL von 255

Tabelle 1: Zusammenfassung der Erkennung	
Meldungen wegen BAF-Schranken	22.428
Gemeldete Dienste	3.324
Tatsächliche Angriffe	278

4 Evaluation des vorgestellten Ansatzes

Zur Beurteilung der Eignung der vorher erwähnten Kriterien haben wir das Suricata IDS dahingehend erweitert, dass die entsprechenden Metriken erfasst und Alarmmeldungen generiert werden. Anschließend wurde eine 6-tägige Messung am Internet-Uplink eines großen Hochschulrechenzentrums durchgeführt. Es wurden alle Pairflows geloggt, die die vorher erwähnten Schwellwerte überschritten haben. Tabelle 1 listet die Haupterkenntnisse auf. Insgesamt wurden hier 22.428 Meldung generiert, die sich auf 3.324 unterschiedliche Dienste bezogen haben. Die Log-Dateien wurden manuell untersucht, um zu entscheiden, ob es sich bei den gemeldeten Pairflows um legitimen Verkehr oder Angriffsverkehr handelt. Hierbei wurden 278 Angriffe identifiziert. Es zeigt sich, dass eine direkte Verallgemeinerung der von Rossow in [Ros14] vorgeschlagenen Erkennungsmethodik auf alle UDP-Ports nicht zum gewünschten Erfolg führt, da nach dieser Methodik alle 22.428 gemeldeten Pairflows als Angriffe zu behandeln wären, obwohl in Wirklich nur 278 bzw. 1.24% der Meldungen sich auf echte Angriffe bezogen haben.

Unter Zuhilfenahme unserer zusätzlichen Kriterien ist eine wesentlich differenzierte Aussage möglich, wobei nicht alle Kriterien eine gleich hohe Selektivität aufweisen: Die Ähnlichkeit bzw. Unähnlichkeit der Payloads hat sich als aussagekräftigstes Kriterium erwiesen. Wie erwartet, weist der Angriffsverkehr hier einen sehr niedrigen Wert auf, während legitimer Verkehr hier deutlich höhere Werte erzielt. Bei den Vergleichen der Paketgrößen stellte sich heraus, dass der Angriffsverkehr zwar ähnliche Größen aufweist, dies aber auch auf einen Teil des legitimen Verkehrs zutriffst, die Aussagekraft dieses Kriteriums ist also geringer als die der Payload-Ähnlichkeit. Das Kriterium der ICMP-Antworten lieferte so gut wie keine Ergebnisse, da wir so gut wie nie ICMP-Antworten beobachten konnten. Ursächlich ist wahrscheinlich, dass die Ressourcen der Opfer so schnell ausgelastet werden, dass keine Möglichkeit mehr zum Senden der ICMP-Pakete besteht. Auch das Ermitteln und Vergleichen der Pfadlängen hat sich als sehr schwer herausgestellt, da eine exakte Messung zu einem nicht mehr reagierenden Rechner schwer ist. Weiterhin ist uns aufgefallen, das teilweise ganze IP-Bereiche „geblackholed“ werden, Pakete an diese Bereiche also bereits an Peering-Points verworfen werden. Insgesamt haben wir festgestellt, dass wir beim Angriffsverkehr nicht in der Lage waren signifikante Unterschiede in der Pfadlänge festzustellen, während es durchaus legitimen Verkehr gab, bei dem wir teil signifikante Unterschiede in der Weglänge feststellen konnten. Dieses Kriterium ist zur Angriffserkennung also eher ungeeignet. Diese Ergebnisse werden in Tabelle 2 zusammengefasst.

Tabelle 2: Aussagekraft der einzelnen Kriterien

Ähnlichkeit der Payloads	+ +
Paketgrößen	+
ICMP Antworten	-
Pfadlängen	- -

5 Abwehr von Amplification Angriffen

Die oben vorgestellte Erkennung von Amplification Angriffen ist jedoch nur sinnvoll in Rechenzentren zu nutzen, wenn diese Erkennung auch zu einer Verhinderung von Angriffen führt, da durchgeführte Angriffe im Allgemeinen Personal und Ressourcen binden, wenn diese von extern gemeldet werden. Dabei müssen zwischen zwei Arten unterschieden werden. Zum einen können die Angriffe reaktiv verhindert werden, so dass beispielsweise nach der Detektion direkt eine Firewallregel erstellt wird, die den weiteren Verlauf eines Angriffs verhindert oder wenigstens abbremst. Auf der anderen Seite gibt es ebenfalls präventive Methoden, die versuchen, eine Schwachstelle zu schließen, bevor sie überhaupt ausgenutzt werden kann.

Beide Möglichkeiten sind im Allgemeinen alleine nicht ausreichend – präventive Methoden sind nur einzusetzen, wenn man einen Angriff erkannt hat, jedoch sollte auch dieser erste Angriff verhindert werden und nur reaktive Methoden helfen nicht, die Systemsicherheit insgesamt zu erhöhen. Aus diesem Grund werden im Folgenden beide Arten näher betrachtet und die einzelnen Vor- und Nachteile gegenübergestellt.

Im Allgemeinen ist den vorgestellten Maßnahmen aber immer ein gut gepflegtes Black- bzw. Whitelisting vorzuziehen, da man nur dann wirklich unter Kontrolle hat, welche Dienste von welchen Nutzern zu welchem Zweck verwendet werden. Dieses in der Theorie und in strikt definierten und abgeschlossenen Netzen gut funktionierende Modell ist jedoch in der Praxis und insbesondere im Hochschulkontext in den Forschungsnetzen nicht durchsetzbar. Vor allem die Internationalität der Nutzergruppe verhindert, dass man das eigene Netz von außen geeignet abschotten kann. Darüber hinaus ist es nicht nur alltägliche Praxis sondern im Allgemeinen auch überaus erwünscht, dass die Wissenschaftler für eigene Zwecke sich und anderen Dienste zur Verfügung stellen, ohne dies im verantwortlichen Rechenzentrum mühsam genehmigen zu lassen. Dass dabei auch immer wieder Fehlkonfigurationen oder veraltete Software zum Einsatz kommen, ist ein notwendiges Übel, das aber unter anderem den Ansatz des Black- bzw. Whitelistings für den Praxiseinsatz untauglich macht.

5.1 Reaktive Maßnahmen

Ist ein auffälliger Verkehrsstrom erkannt, so sollte man versuchen, den weiteren Angriff zu verhindern bzw. einzudämmen. Da eines der Charakteristika einer Amplification Attacke ist, dass diese einen regulären Dienst verwendet, ist die Abwehr des Angriffs nicht trivial durchzuführen.

Die offensichtlichste Abwehrmaßnahme ist die Nutzung von Firewalls, die entweder basierend auf Ports oder IP-Adressen verhindert, dass verdächtige Anfragen eintreffen bzw. nach außen gesendet werden. Diese radikale Möglichkeit verhindert, dass durch den verwundbaren Dienst

weiterer Schaden angerichtet werden kann. Dabei ist jedoch zu bedenken, dass Kollateralschäden der Maßnahme unvermeidlich sind. Insbesondere wenn Angreifern diese Abwehrstrategie bekannt ist, kann dies zielgerichtet zum eigenen Schaden als DoS-Attacke missbraucht werden.

Auch wenn man sich dieses Risikos bewusst ist, ist es wichtig, zu entscheiden, ob die Firewallregel den eingehenden oder den ausgehenden Verkehr blockieren soll. Blockiert man den ausgehenden Verkehr, so stoppt man damit augenscheinlich den Angriff auf das Opfer-System, da keinerlei Datenpakete mehr ab diesem Zeitpunkt weitergeleitet wird. Jedoch werden dadurch die einkommenden Datenpakete logischerweise nicht abgeblockt, was zur Folge hat, dass der Amplifier die Anfragen bearbeitet und weiterversendet. Dies stellt zwar im Allgemeinen kein großes Problem dar, aber kann, falls zu viele Amplification Angriffe durchgeführt werden oder andere Engpässe auftreten, zu Performanceeinbußen führen.

Beim Blockieren des eingehenden Verkehrs entlastet man zwar den diensteerbringenden Server, da dieser keine Anfragen mehr vom vermeintlichen Angreifer erhält, jedoch sind es in diesem Fall – je nach Platzierung der Firewall gegenüber der Analyseeinheit – im Allgemeinen noch die Firewall-Logs auszuwerten, damit man erkennen kann, wann der Angriff beendet ist. Weiterhin kann es sein, dass die Sperrung des eingehenden Verkehrs zu spät erfolgt, da unter Umständen die Verarbeitung auf dem Amplifier einige Zeit in Anspruch nimmt und somit der Großteil der Anragen schon durch die Firewall durchgeroutet wurde.

Neben der kompletten Sperrung von IP-Adressen oder Ports bietet es sich an, eine Bandbreitenlimitierung einzusetzen. Diese funktioniert nach dem gleichen Prinzip wie die Sperrung, d. h. es ist möglich, sowohl den eingehenden als auch den ausgehenden Verkehr zu drosseln. Auch bei den Vor- bzw. Nachteilen entspricht diese Maßnahme den oben genannten. Vorteil der Bandbreitenlimitierung ist jedoch, dass der Dienst nicht vollständig geblockt wird und somit ein Missbrauch der Schutzmaßnahme unwahrscheinlicher wird.

Auf der anderen Seite muss die Bandbreitenlimitierung so ausgelegt werden, dass sich die Amplification-Angriffe nicht mehr lohnen. Der oben eingeführte Bandwith Amplification Factor (BAF) gibt das Verhältnis zwischen der Größe des Payloads gesendet zum Server und gesendet zum Client an. Hat nun eine spezifische Amplification Attacke einen einen hohen BAF-Wert, so muss die Limitierung der Bandbreite entsprechend höher als dieser Faktor sein, ansonsten ist der Amplification Angriff trotz der Gegenmaßnahme zwar mit verminderter Stärke erfolgreich.

Schlussendlich wird, wenn die Limitierung nicht spürbar ist, das Opfer unter Umständen auf die Beteiligung an dem Angriff hinweisen, so dass eine manuelle Nachbearbeitung trotz allem immer noch nötig ist.

5.2 Präventive Maßnahmen

Um gegen Amplification Angriffe gewappnet zu sein, ist es deshalb unerlässlich, die Angriffe schon abzuwehren, bevor sie ausgenutzt werden können. Diese präventiven Maßnahmen können natürlich nur angewandt werden, wenn man sowohl die eigene Infrastruktur bzw. die eigenen Dienste kennt, als auch wenn man die verwundbaren Dienste bzw. Protokolle ermittelt hat.

Das Ermitteln der eigenen Dienste bzw. der Dienste, die innerhalb des eigenen Forschungsnetzes betrieben werden, kann beispielsweise mit der in [vMH13] beschriebenen Methode durchgeführt werden. Die dafür nötigen Portscans sind auf der einen Seite relativ schnell und problemlos durchzuführen, aber auf Nutzerseite werden diese Portscans im Allgemeinen nicht gerne gesehen, da durch die Nutzung von Versions- bzw. Diensteerkennung zumeist sehr viele unnötige Einträge in Logdateien provoziert werden. Daher ist es üblich, dass einige Systeme und Dienste nicht zuverlässig erkannt werden können, da unter Umständen die scannende Maschine auf einer lokalen Blackliste steht.

Ohne die genaue Kenntnis ist jedoch, wie schon oben beschrieben, ein Black-/Whitelisting-Ansatz nicht durchführbar. Erschwerend kommt hinzu, wenn Dienste nicht auf den Standardports betrieben werden und somit beispielsweise von einer Portsperre von der regulären Nutzung ausgeschlossen wären. Analog verhält es sich mit Bandbreiten- oder Paketlimits, die die Nutzung von regulären Diensten einschränkt, ohne dass diese Einschränkung in einem sinnvollen Verhältnis zum gewonnenen Schutzniveau beitragen würde.

Somit ist es schwierig, die oben beschriebenen reaktiven Maßnahmen präventiv einzusetzen. Ziel muss daher sein, für jedes betroffene System bzw. für jeden betroffenen Dienst eine sichere Konfiguration bereitzustellen. Zur Prävention wird somit ein mehrstufiges Verfahren eingesetzt.

In einem ersten Schritt wird mit Hilfe des in diesem Paper vorgeschlagenen Verfahren festgestellt, dass eine Amplification Attacke durchgeführt wird bzw. wurde. Dabei werden insbesondere die von extern zugesandten Beschwerdeschreiben als Bestätigung der True-Positiv-Erkennung herangezogen.

In dem darauffolgenden Schritt wird ermittelt, um welchen Dienst bzw. um welches Protokoll es sich bei dem Angriff gehandelt hat. Ist dies ermittelt, wird zum einen untersucht, wie eine sichere Konfiguration auszusehen hat. Diese kann entweder darin liegen, dass Software-Updates durchgeführt werden, wenn beispielsweise eine Applikation fehlerhafterweise zu viele Daten auf eine Anfrage zurückmeldet, es können auch Konfigurationsänderungen nötig werden, die beispielsweise verhindern, dass auf bestimmte Anfragen geantwortet wird, oder es kann – gerade bei Schwächen im zugrundeliegenden Protokoll – nötig sein, dass für den speziellen Dienst oder das System ein individueller Black-/Whitelisting-Ansatz implementiert wird.

Zum anderen wird ebenfalls ermittelt, wie man diese Schwachstelle ggf. austesten kann. Diese Tests auf Schwachstellen sollten nun in den regelmäßigen Scans des eigenen Netzes mit aufgenommen werden. Da üblicherweise Amplification Angriffe sich nicht alle potentiell verwundbaren Systeme ausnutzen, sind in den meisten Fällen noch weitaus mehr betroffene Systeme im eigenen Netz vorhanden.

Schlussendlich wird wieder beim ersten Schritt begonnen, sobald der nächste Angriff detektiert wurde. Im Allgemeinen wird man dort keine neuen oder neuartigen Angriffe sehen können, so dass die Suche nach geeigneten Gegenmaßnahmen auf schon bekanntes zielen wird. Durch die regelmäßige Fluktuation im Hochschulumfeld ist es jedoch nicht ungewöhnlich, dass bereits bekannte Fehler erneut gemacht werden.

Die präventiven Maßnahmen reduzieren sich somit auf der Herausforderung, schneller die eigenen Systeme und Schwachstellen zu erkennen bzw. zu schließen, bevor ein Angreifer sie

ausnutzen kann. Die vorgestellte Methode in diesem Paper ermöglicht es aber auch, bisher unbekannte Angriffe zu erkennen und schlussendlich zu verhindern.

6 Zusammenfassung und Ausblick

Diese Arbeit zeigt, mit welchen Mitteln eine Erkennung von Amplification Angriffen innerhalb des Netzes des Amplifiers möglich ist. Dabei wurde als Grundlage eine schon existierende Betrachtung von Amplification Angriffen verwendet, die durch verschiedene weitere Kriterien deutlich verbessert werden konnte. Dabei wurden eine Reihe von verschiedenen möglichen Kriterien ermittelt und auf der einen Seite theoretisch untersucht und auf der anderen Seite praktisch evaluiert. Dabei haben sich einige Kriterien als zielführender als andere erwiesen. Insbesondere das Kriterium der fehlenden Entropie des Payloads hat sich als sehr effektiv herausgestellt, da ein Angreifer im Allgemeinen viele ähnliche Anfragen stellt, die dann dementsprechend in ähnlichen Antworten münden.

Da es jedoch meist nicht damit getan ist, die Amplification Attacken nur zu erkennen, wurden auch verschiedene Methoden der Reaktion bzw. Prävention diskutiert. Dabei ergab sich, dass die Reaktion zwar ein probates Mittel ist, eine laufende Attacke wenigstens abzumildern, jedoch eine umfassende Prävention dieser in jedem Fall vorzuziehen ist.

Gerade in diesem Bereich gibt die vorliegende Arbeit wertvolle Hinweise, indem neue Angriffsvektoren bzw. unbekannte Amplification Angriffe schnell und präzise erkannt werden können.

Die in dieser Arbeit gewonnenen Erkenntnisse sollen nun in einem nächsten Schritt dafür verwendet werden, neue Protokolle zu identifizieren, bei denen ebenfalls ein Amplification Angriff möglich ist. Weiterhin ist noch zu untersuchen, wie eine adaptive Anpassung der Schwellenwerte für die einzelnen Kriterien realisiert werden können. Bisher wurden die Schellenwerte für das Szenario eines großen deutschen Hochschulrechenzentrums ausgelegt, jedoch ist dadurch noch nicht gesagt, dass diese Kriterien universell in jeder Umgebung einsetzbar sind.

Insbesondere bei der Erkennung neuer Angriffsvektoren ist eine Variation der Schwellenwerte unter Umständen sehr sinnvoll, um nutzbare Ergebnisse aus der Analyse ziehen zu können.

Schlussendlich muss erörtert werden, wie die Prävention besser durchgeführt werden kann, so dass beispielsweise bei der Erkennung eines neuen Dienstes durch einen Portscan gleich ein Vulnerability-Scan durchgeführt wird, der in regelmäßigen Abständen wiederholt wird, wie dies unter anderem [vMH13] vorsieht.

Danksagung

Teile dieser Arbeit wurden durch das Bundesministerium für Bildung und Forschung gefördert (FKZ: 16BP12309). Die Autoren danken den Mitgliedern des Munich Network Management Teams (MNM-Team) für wertvolle Kommentare zu früheren Versionen dieses Artikels. Das MNM-Team ist eine Forschungsgruppe der Münchener Universitäten und des Leibniz-Rechenzentrums der Bayerischen Akademie der Wissenschaften unter der Leitung von Prof. Dr. Dieter Kranzlmüller und Prof. Dr. Heinz-Gerd Hegering.

Literatur

[CER98] CERT: *Smurf IP Denial-of-Service Attacks CA-1998-01*. http://www.cert.org/historical/advisories/CA-1998-01.cfm, Januar 1998.

[Deu96] DEUTSCH, P.: *DEFLATE Compressed Data Format Specification version 1.3*. RFC 1951 (Informational), Mai 1996.

[Fat13] FATIH ÖZAVCI: *VOIP Wars: Return of the SIP*. DEFCON 21 - http://www.defcon.org/images/defcon-21/dc-21-presentations/Ozavci/DEFCON-21-Ozavci-VoIP-Wars-Return-of-the-SIP-Updated.pdf, August 2013.

[KMGG07a] KAMBOURAKIS, GEORGIOS, TASSOS MOSCHOS, DIMITRIS GENEIATAKIS und STEFANOS GRITZALIS: *A Fair Solution to DNS Amplification Attacks*. In: *Second International Workshop on Digital Forensics and Indicent Analysis (WDFIA 2007)*. IEEE, August 2007.

[KMGG07b] KAMBOURAKIS, GEORGIOS, TASSOS MOSCHOS, DIMITRIS GENEIATAKIS und STEFANOS GRITZALIS: *Detecting DNS Amplification Attacks*. In: *Critical Information Infrastructures Security (CRITICS 2007)*. Springer, 2007.

[Pri13] PRINCE, MATTHEW: *The DDoS That Almost Broke the Internet*, März 2013. last accessed: June 2014.

[Ros14] ROSSOW, CHRISTIAN: *Amplification Hell: Revisiting Network Protocols for DDoS Abuse*. In: *Proceedings of the 2014 Network and Distributed System Security (NDSS) Symposium*, San Diego, CA, Februar 2014.

[RSR09] RASTEGARI, SAMANEH, M IQBAL SARIPAN und MOHD FADLEE A RASID: *Detection of Denial of Service Attacks against Domain Name System Using Neural Networks.* International Journal of Computer Science Issues (IJCSI), 7(4), 2009.

[SL02] SPECHT, STEPHEN und RUBY LEE: *Distributed Denial of Service: Taxonomies of Attacks, Tool and Countermeasures*. In: *Proceedings of the ISCA 17th International Conference on Parallel and Distributed Computing Systems*, San Francisco, CA, September 2002.

[SLS08] SUN, CHANGHUA, BIN LIU und LEI SHI: *Efficient and Low-Cost Hardware Defense Against DNS Amplification Attacks*. In: *IEEE Global Telecommunications Conference 2008. (GLOBECOM 2008)*. IEEE, 2008.

[Sol14] SOLUK, KIRK: *NTP ATTACKS: Welcome to The Hockey Stick Era*. http://www.arbornetworks.com/asert/2014/02/ntp-attacks-welcome-to-the-hockey-stick-era/, Februar 2014.

[vMH13] VON EYE, FELIX, STEFAN METZGER und WOLFGANG HOMMEL: *Dr. Portscan: Ein Werkzeug für die automatisierte Portscan-Auswertung in komplexen Netzinfrastrukturen*. In: PAULSEN, CHRISTIAN (Herausgeber): *Sicherheit in vernetzten Systemen: 20. DFN Workshop*, Seiten C-1–C-21, Norderstedt, Deutschland, Januar 2013. Books on Demand.