فصلنامهٔ پدافند غیرعامل و امنیت (پژوهشی)
سال دهم: شمارهٔ ۳۸- بهار ۱۴۰۱
صص: ۳۱ - ۵۸
تاریخ دریافت: ۱۴۰۰/۱۲/۸ تاریخ پذیرش: ۱۴۰۱/۲/۶

Quarterly of Passive Defense & Security
Vol.10, No.38,Spring 2022
P-P: 31 - 58

ارائه راهکار برای مقابله با حملات انعکاسی و تقویت شده DoS در سرویس Bind9

مهدی محمدخانی[۱]
محمدعلی دوستاری[۲]

چکیده:

در این پژوهش با بهره گیری از پروتکل‌های مورد استفاده در سرویس نام دامنه و مسیریابی شبکه، راهکاری برای مقابله با حملات تقویت‌شده و انعکاسی که از انواع حملات منع خدمت هستند، ارائه شده است. این راهکار در سرویس نام دامنه امکانی را فراهم می‌کند تا بتوان بسته درخواست نام دامنه که توسط یک کلاینت در شبکه ایجاد و ارسال شده است را در سرور نام دامنه بررسی نمود و در صورتی که بسته از طرف یک حمله‌کننده صادر شده بود، شناسایی شده و از ایجاد یک حمله که می‌تواند موجب از کار افتادن سرویس نام دامنه بشود، جلوگیری کند. این راهکار برروی بستر مجازی ساز در سرویس نام دامنه سیستم عامل لینوکس ورژن ۹، انجام‌شده و نتایج آن ثبت و ضبط شده است. با به‌کارگیری این روش برروی سرویس سیستم عامل لینوکس ورژن ۹ بیش از هفتاد درصد از حملات منع خدمت تقویت شده و انعکاسی که با هدف از کار انداختن سرویس نام دامنه طراحی و ایجاد شده اند شناسایی و با آنها مقابله شده است.

کلیدواژه: حملات DoS، سرویس DNS، حملات انعکاسی، حملات تقویت شده.

۱. دانشجوی کارشناسی ارشد معماری سیستم های کامپیوتری، دانشگاه شاهد، نویسنده مسئول *Mohammadkhani4898@hotmail.com*
۲. دانشیار دانشگاه شاهد، گروه مهندسی کامپیوتر *Dostari@shahed.ac.ir*

۱. مقدمه

از آنجایی که درخواست های سرویس DNS[1] از نوع درخواست‌های UDP[2] هستند، امکان اعتبارسنجی[3] آدرس‌های مبداء موجود در بسته درخواست های DNS وجود ندارد، به همین دلیل حمله کننده می تواند د استهایی با آدرس جعلی و یا با آدرس نود[4] دیگری در شبکه که قصد حمله به آن را دارد تولید کرده و برای سرویس DNS ارسال کند. در اصطلاح به این نوع از حملات، حملات انعکاسی[5] می گویند. تنها در صورتی می توان با این نوع از حملات مقابله نمود که شرکت های ارائه کننده خدمات اینترنت بسته‌ها را براساس آدرس‌های مبداء فیلتر کنند. اما متاسفانه به دلیل این که بار[6] بسیار بالایی در تجزیه و تحلیل این روش وجود دارد همه شرکت‌های ارائه کننده خدمات اینترنت قادر به انجام این کار نمی باشند.

حمله کننده همچنین می‌تواند درخواست های کوچکی به سمت سرویس DNS ارسال کند و در پاسخ بسته‌هایی با طول بسیار بیشتر توسط سرویس DNS ایجاد شود که این کار ایجاد بار بالای سرور DNS را در پی هد داشت. به عنوان مثال حمله کننده می‌تواند درخواست‌هایی با طول ۳۰ بایت را برای سرور ارسال کند و در جواب سرویسی که مورد هدف قرار گرفته است پاسخی با طول ۳۰۰۰ بایت ایجاد کند. به این نوع از حملات اصطلاحاً حملات تقویت شده[7] گویند[۳].

به جهت مقابله با این نوع حملات راهکارهایی در دنیا ارائه شده است که متاسفانه یا به کارگیری آنها بسیار سخت است و یا در صورت به کارگیری برروی کاربران قانونی سرویس DNS تاثیرگذار خواهد بود؛ بدین معنا که درروند دریافت پاسخ اینگونه کاربران اختلال ایجاد خواهد شد. راهکاری که در این پژوهش به کار گرفته شده است برروی سطوح بالایی دامنه[8] کاربرد دارد.

همان طور که گفته شد در این گونه حملات برخلاف پروتکل TCP[9] دیگر ارتباط اولیه‌ای[10] صورت نمی پذیرد و سرویس DNS به مجرد اینکه که درخواستی را دریافت کرد آن را پردازش می کند و جواب را به IP[11] مبدا ارسال می کند.

۲. مرورادبیات

در ابتدا به منظور درک بهتر موضوع به تعریف اصطلاحات تخصصی به کار رفته در این مقاله می‌پردازیم:

1. *Domain Name System*
2. *User Datagram Protocol*
3. *V lidation*
4. *Node*
5. *Reflection Attacks*
6. *Load*
7. *Amplification*
8. *Top-Level Domain*
9. *Transmission Control Protocol*
10. *Handshake*
11. *Internet Protocol*

۱/۲. اصطلاحات:

۱/۱/۲. رکورد NS:

رکوردی است که به نام یک سرور اصلی که مالک و سازنده یک ناحیه است، اشاره می‌کند. به عبارت دیگر هر ناحیه توسط یک سرور ساخته شده است و یک رکورد وجود دارد که به نام آن سرور اشاره می‌کند[۱۸].

۲/۱/۲. عملیات Probe:

عملیات *Probe* یا به طور کامل *Network Probe*، عملیاتی است که در آن حمله‌کننده سعی می‌کند به صورت گسسته بسیاری از امکانات و تجهیزات را کنار[1] بگذارد و به اطلاعاتی از قبیل نام شرکت تولید کننده تجهیزات امنیتی که مورد استفاده در شبکه، اطلاعات لایه های انتقال، نشست، ارائه و برنامه کاربردی[2]، اطلاعاتی که درون فایل ها ذخیره شده است (می تواند تنظیمات سیستم های یونیکسی باشد) و ... دست پیدا کند. هدف از جمع آوری این اطلاعات توسط حمله کننده‌ها این است که بتوانند با در دست داشتن اطلاعات بیشتر از هدف، حمله ای گسترده تر و اثرگذارتر را از خود بجای بگذارند[۱۲].

۳/۱/۲. کنسرسیوم سیستم های اینترنتی [3]

کنسرسیوم سیستم‌های اینترنتی یک موسسه غیرانتفاعی در زمینه تکنولوژی اینترنت است. این موسسه به وسیله تولید نرم افزارها و پروتکل‌های مهم و پایه‌ای در اینترنت از زیرساخت جهانی پشتیبانی می کند. یکی از کارهایی که این شرکت انجام داده است، پیاده‌سازی تکنولوژی‌هایی بر بستر اینترنت از قبیل *BIND* (منظور سرویس DNS است که سیستم عامل لینوکس درحال حاضر از نسخه ۹ آن استفاده می‌کند.) و *ISC DHCP* و *ISC AFTR* است.

یکی از سیزده سرور DNS ریشه که در جهان فعالیت می کند متعلق به موسسه ISC است. این موسسه همچنین پس از گذشت چندین سال، فعالیت خود را در حوزه های پروژه‌ای نیز گسترش داده و محصولاتی از قبیل *NetBSD* و *Xfree* را ایجاد کرده است.

۴/۱/۲. نام واجد شرایط[4] [۱۷]

نام واجد شرایط یا به اختصار *Qname* به یک نامی گفته می‌شود که بیانگر و توصیف گر یک موجودیت در شبکه است. این نام، هویت، خصوصیات و صفات المان را بیان می‌کند[۱۴].

1. *Bypass*
2. *The fourth, fifth, sixth and seventh layers in OSI Model*
3. *Internet Systems Consortium (ISC)*
4. *Qualified Name*

۵/۱/۲. سیستم نام دامنه [1]

سیستم نام دامنه که به اختصار DNS گفته می‌شود، به صورت کلی و اجمالی وظیفه مدیریت، نظارت و بررسی نام‌های موجود در شبکه را برعهده دارد[۱۵].

۶/۱/۲. حملات منع خدمت[2]

این گونه از حملات به منظور از کار انداختن سرویسی که در شبکه مورد استفاده قرار می‌گیرد، اجرایی می شود. اینگونه حملات دارای انواع و روش های مختلفی می‌باشد و با استفاده از روش های مختلف تلاش می کند تا روند کار سرویس مدنظر حمله کننده را مختل کنند[۲۲].

۷/۱/۲. درخواست[3]

منظور از درخواست، همان تقاضایی است که یک کلاینت از یک سرور در شبکه دارد. بصورتی که درخواست خود را در قالب بسته‌ای برای سرور ارسال می کند[۸].

۲/۲. پیشینه تاریخی پژوهش‌های صورت گرفته:

در صنعت فناوری اطلاعات راهکارها و قابلیت هایی برای مقابله با این نوع از حملات وجود دارند که می توانند تا حدودی از شدت آنها بکاهند؛ در این زمینه می توان به راهکارهایی از قبیل استفاده از دیوار آتش۴، هدایت حملات به سمت بخش‌هایی از شبکه به این منظور که بار زیادی بر شبکه تحمیل نشود و سیستمها بتوانند به درستی به وظایف خود عمل کنند؛ استفاده از کنترل کننده شبکه تعریف نرم افزار، استفاده از سرویس های تشخیص و حفاظت نفوذ۵، فیلتر کردن بسته‌های ورودی، از بین بردن درخواست‌های «هرکدام»، روش *D-WARD* [۱۹]، روش *DAAD*[۶]و... اشاره نمود.

هیچ کدام از راهکارهای ارائه شده نمی توانند به صورت کامل با انواع و اقسام حملات DoS مقابله کنند. البته شایان ذکر است که این نکته دلیل بر عدم کارایی و عملکرد اینگونه روش‌ها نمی باشد بلکه از آنجایی که هر کدام از روش های بیان شده تنها می توانند الگوهای خاصی از اینگونه حملات را شناسایی کنند، یک روش، برای مقابله با تمامی روش‌ها کارایی لازم و کافی را نداشته است.

در بخش بعد به بیان روش‌های مقابله با اینگونه حملات که در دنیای آکادمیک و صنعت پیش بینی شده است، نقاط ضعف و دلایل عدم کفایت روش‌ها در جهت مقابله با حملات DoS پرداخته شده است.

۱. استفاده از دیوار آتش ماشین میزبان:

سرویس‌های DNS نیز همانند سایر سرویس‌های شبکه می بایست برروی یک سرور فیزیکی و یا مجازی پیاده‌سازی شوند. از طرف دیگر سرورهای فیزیکی به تنهایی دارای یک دیوارآتش نرم افزاری است که

1. *Domain Name System*
2. *Denial of Service (DoS)*
3. *Query*
4. *Firewall*
5. *Intrusion Detection System & Intrusion Protection System (IPS/IDS)*

اصطلاحا به آن دیوارآتش ماشین میزبان گفته می شود[۷]. نقاط ضعف استفاده از این دیوارهای آتش در مقابله با حملات منع خدمت به شرح ذیل می باشد:

۱/۱. دیوارهای آتش میزبان عموما دارای قوانین[1] ساده ای هستند و توانایی مقابله با حملات گسترده و پیچیده‌ای که امروزه سیستم‌ها را تهدید می کنند را ندارند.[۴].

۲/۱. دیوارهای آتش میزبان در لایه سوم مدل لایه ای OSI[2] کار می‌کنند و به همین جهت توانایی تشخیص حملاتی که به صورت کد مخرب در قسمت پیام[3] بسته ها قرار دارند را ندارند. به عبارت دیگر این دیوارهای آتش تنها به پروتکل و نوع ترافیک توجه می‌کنند[۱۰].

۳/۱. به دلیل اینکه این دیوارهای آتش برروی سرورهای فیزیکی قرار دارند که آن سرورها پذیرای سرویس‌ها و وظایف بزرگ دیگری نیز هستند، باعث می شود که سربار[4] آن سرور زیاد شده و حمله کنندگان بتوانند با بالا بردن حجم کاری[5] آن سرور، دیوارهای آتش آن را از کار بیاندازند[۲۱].

۲. هدایت حملات به سمت مکان های دیگر

چهار شخص به نام های *Takashi Ikegawa*، *Ryoichi Suzuki*، *Takafumi Hamano* و *Hiroyuki Ichikawa*، مقاله‌ای از طرف آزمایشگاه *NTT Network Service Systems* مبنی بر ارائه راهکاری برای جلوگیری از حملات DoS را منتشر کردند.

در این مقاله تمرکز برروی شرکت های ارائه‌کننده خدمات اینترنتی[6] است که آن‌ها بتوانند در این حملات به کاربران و مدیران سرویس‌ها کمک کنند. در این مقاله راهکار انتقال بسته ها به مقاصد دیگر ارائه شده است.

روشی که در این مقاله ارائه شده است بدین صورت است که پس از آنکه این نوع حملات توسط شرکت‌های ارائه دهنده خدمات اینترنتی شناسایی شد به سمت روترها و گره‌هایی انتقال پیدا کند که حاشیه‌ای هستند و به عبارتی برای همین کار تعبیه شده‌اند و در آنجا در یک حلقه که این شرکت ها تعبیه کرده ی بمانند تا از بین بروند و به سیستم‌ها و شبکه‌های کاربران راه پیدا نکنند[۱۳].

نقاط ضعف این روش به شرح زیر است:

۱/۲. شرکت های ارائه دهنده خدمات اینترنتی متحمل هزینه‌های بسیاری می‌شوند زیرا می بایست تجهیزات سخت افزاری مازادی را تعبیه کنند تا بتوانند ترافیک های مخرب را به سمت این تجهیزات هدایت کنند.

۲/۲. در صورتی که این شرکت‌ها بخواهند حلقه هایی را در این قسمت ایجاد کنند می بایست تجهیزات

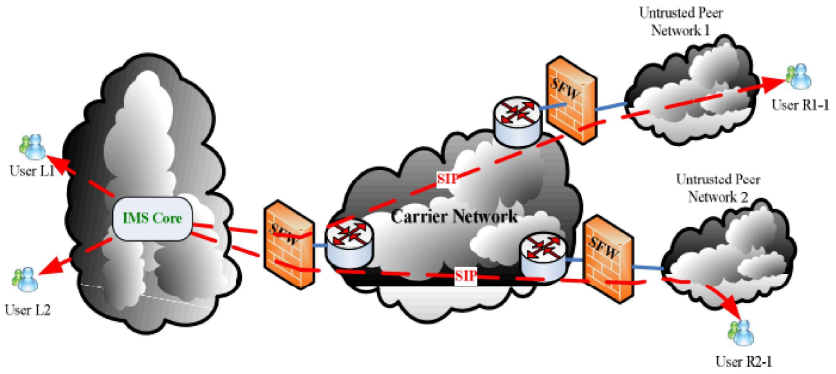
1. *Rules*
2. *Open Systems Interconnection*
3. *Payload*
4. *Header*
5. *Work Load*
6. *ISP*

بیشتری را خریداری کرده تا بتوانند بسته‌هایی را که به این قسمت هدایت می‌شوند را به وسیله تاریخ انقضاء از بین ببرند.

۳/۲. با بکارگیری این روش میزان پهنای باند بسیار زیادی از شرکت‌های ارائه دهنده خدمات اینترنتی اشغال می‌شود و همین امر موجب می‌شود که این شرکت‌ها نتوانند به مشتریان خود سرویسی با کیفیت بالا ارائه دهند و در نهایت مشتریان خود را از دست بدهند و هزینه زیادی را متضرر بشوند.

۳. استفاده از کنترل کننده شبکه تعریف نرم افزار

در سال ۲۰۱۴ راهکارهایی در قالب یک مقاله بیان شده است که سیستم‌هایی را که در فضای ابری هستند در مقابل حملاتی که با خدشه وارد کردن به قوانین لایه پنجم مدل لایه‌ای OSI قصد خرابکاری دارند، ایمن سازند.



کل ۱ چندلایه سازی بکه

همان‌طور که د کل ۱ مشاهده می‌شود در این مکانیزم شبکه به چندین قسمت مختلف تقسیم می‌شود و در مابین هر قسمت از تجهیزات امنیتی استفاده می‌شود. در این روش از مدل صف دارای اولویت استفاده شده است به صورتی که تمامی بسته ایی که می‌خواهند وارد فضای ابری شوند دارای یک مقداری هستند به نام ارزش اولویت و طبق آن می‌توانند وارد شبکه داخلی شوند[۱۶].

کل ۱ یک سری تجهیزات سخت افزاری به نام کنترل کننده شبکه تعریف نرم‌افزار (Software-Defined Networking) یا SDN در ناحیه غیرنظامی قرار می‌دهند تا بتوانند ترافیک هایی که قرار است به قسمت‌های مختلف برود را از نظر اینکه تهدیدی برای تجهیزات نرم افزاری است یا خیر بررسی و تحلیل کند[۱].

نقاط ضعف این روش به شرح زیر است:

۱/۳. به دلیل آنکه این روش از مکانیزم صف اولویت دار استفاده می‌کند و می بایست روترها بتوانند بسته‌های بیشتری را نسبت به قبل در حافظه خود جای دهند نیاز به بافر دارند. اضافه کردن بافر درون روترها موجب به افزایش مساحت روترها و افزایش قیمت روترها می‌شود.

۲/۳. این مکانیزم نیازمند تجهیزات گسترده و گران قیمت است که ممکن است هر شرکت و سازمانی نتواند آن‌ها را تهیه کند.

۳/۳. در این روش تهدیدات و حملاتی شناسایی می‌شود که تنها نرم افزارهای موجود در سازمان را مورد هدف قرار می‌دهد.

۴/۱. این مکانیزم به دلیل اینکه تنها بسته هایی که از خارج از سازمان وارد سازمان می‌شوند را بررسی می‌کنند، نمی‌توانند حملات *DoS Reflector* را شناسایی کنند و به طبع آن نمی‌توانند با این‌گونه حملات مقابله کنند.

۴. استفاده از سرویس های تشخیص و حفاظت نفوذ [1]

در استفاده از سیستم های تشخیص نفوذ، مدیران سیستم می‌بایست به صورت دستی اقدام به مسدود سازی ترافیک ها کنند. درحالی که در این سیستم ها در کنار این موضوع که به مدیر سیستم هشدار داده می‌شود، به صورت اتوماتیک خود سیستم این نوع ترافیک‌ها را مسدود می‌کند. نمونه ای از این سخت افزار را د شکل ۲ می توانید مشاهده کنید. این سخت افزار تولید شرکت *Juniper* می‌باشد[۲۰].

شکل ۲ سیستم تشخیص و حفاظت از نفوذ شرکت *Juniper*[2]

نقطه ضعفی که این نوع دستگاه دارد این است که توانایی مقابله با حملات خودداری از خدمت توزیع شده[3] را ندارد.

۵. فیلتر کردن بسته های ورودی

در این روش بسته هایی که دارای آدرس های مبداء جعلی و دستکاری شده می باشند، در ابتدای ورود

1. *Intrusion Detection System & Intrusion Protection System (IPS/IDS)*
2. *Juniper IDP800*
3. DDoS

به شبکه و زمانی که هنوز به سرورهای حساس نرسیده باشند، فیلتر می‌شوند. این کار بیشتر در شرکت های ارائه کننده خدمات اینترنت صورت می پذیرد. بدین صورت که برای مشتریان هر ISP یک رنجی مشخص شده است و ISP می تواند در صورتی که بسته‌ای خارج از آن رنج IP روئیت شد، آن را فیلتر کند. این روش تنها می تواند با حملات انعکاسی مقابله کند و نمی تواند با حملات تقویت شده [۱۱] مقابله کند. البته با حملات انعکاسی که در همان رنج IP است نیز نمی تواند مقابله کند. پس نتیجه این می شود که این روش تنها بخش محدودی از حملات را می تواند شناسایی کند و با آنها مقابله کند. این کار معمولا در مسیریاب های[1] شرکت های ارائه کننده اینترنت به کارگیری می شود و ارتباطی به ادمین شبکه سازمان ندارد[۵].

۶. از بین بردن درخواست های هرکدام

در این روش طبق توضیحی که داده شد برای جلوگیری از روی دادن حملات تقویت شده از رسیدن درخواست هایی از نوع «هرکدام» به سرویس DNS خودداری می کنیم. برای انجام این کار می بایست برروی لبه شبکه در دیوار آتش قوانینی را پیاده سازی کنیم. البته این کار می تواند دو نقطه ضعف داشته باشد.

۱٫۶. با انجام این کار موجب می شود به اشتباه درخواست‌هایی که از سمت کاربران حقیقی سرویس DNS صادر شده است بسته شود و دیگر آنها نتوانند اطلاعات کاملی را مرتبط با یک نام دامنه از سرویس DNS دریافت کنند.

۲٫۶. با انجام این کار و رشد استفاده از روش سرویس DNS ایمن و افزایش استفاده از رکوردهای RRSIG و DNSKEY و TX به راحتی حمله کننده می واند به استفاده از آنها بپردازد و حملات سنگین تری را ارسال کند.

۷. راهکارهای نوین به کارگیری شده در صنعت فناوری اطلاعات

یکی از مواردی که برای جلوگیری از این نوع حملات، گفته می‌شود، استفاده از سرور DNS ایمن[2] است. اما، جالب است بدانیم، این ویژگی نه تنها با این نوع از حملات مقابله نمی کند، بلکه، باعث می شود، درصورت قرارگیری در معرض این نوع حملات، آسیب‌های بیشتر به سازمان وارد شود. دلیل آن این است که، این ویژگی برای این که امنیت بسته ها را تأمین کند، باید قسمت های دیگری، مازاد بر قسمت های پاسخی که سرور DNS در حالت عادی می‌دهد، اضافه کند. به همین دلیل، در صورتی که، تحت حملات تقویت‌شده قرار گیرد، می تواند حدوداً ۵۰ برابر بسته های پاسخ در حالت عادی، بسته تولید کند که این امر موجب تحمیل بار بیشتر به سرور DNS می‌شود[۲].

1. *Routers*
2. DNSSEC

۳. یافته‌های تحقیق

در روش ارائه شده در این مقاله درزمانی که یک پاسخ قرار است متناسب با درخواست دریافت شده، ایجاد شود، سرور DNS یک نامی را به زیرشبکه ای که کلاینت در آن قرار دارد، اختصاص می‌دهد. این نام می تواند یکی از موارد زیر باشد:

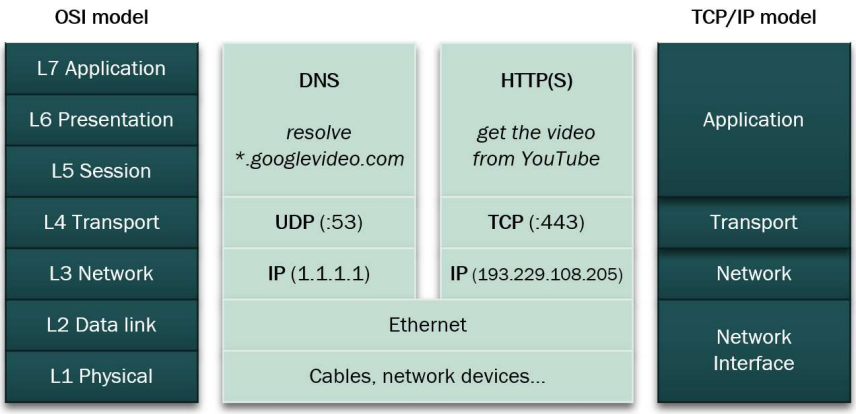
۱. نام یک *Wildcard*

۲. نام کامل ارسال کننده درخواست[1]

۳. نام منطقه و ناحیه[2]

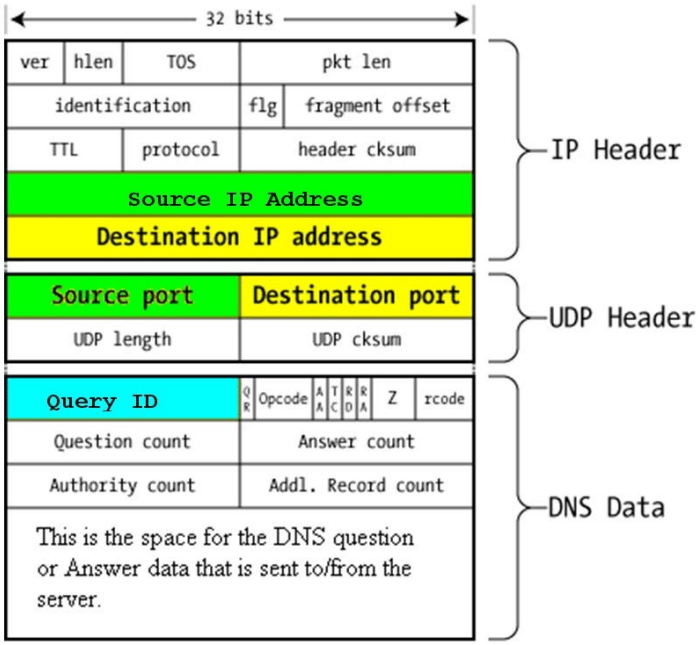
امروزه مهم ترین معیار و ساختار بررسی بسته‌های منتقل شده در شبکه، مدل لایه ای OSI می باشد؛ این مدل د شکل ۳ نمایش داده شده است.

همان طور که در فصول پیشین بیان شد، اتفاقی که در حملات انعکاسی روی می‌دهد این است که حمله کننده به جای آن که در قسمت آدرس *IP* مبداء، آدرس *IP* خود را قرار دهد، آدرس یک نود از شبکه که قصد حمله به آن را دارد، قرار داده تا در زمانی که بسته ارسالی توسط سرویس DNS پردازش شد، سرور DNS بسته های پاسخ را به سمت آن سیستم ارسال کند. حال اتفاقی که می‌بایست روی دهد این است که باید به نحوی سرویس DNS متوجه صحت آدرس مبداء بشود.



شکل ۳ شیوه قرارگیری بسته DNS در مدل *OSI*

1. *QNAME*
2. *Zone Name*



شکل ۴ نمونه ای از بسته ارسالی DNS

درحالتی که سیستم DNS تحت حمله قرار ندارد می توان مکان قرارگیری سیستم های حاضر در شبکه را متوجه شد؛ این کار بدین صورت انجام می شود که در زمان های عادی (منظور زمان هایی است که سیستم و شبکه تحت حمله قرار نگرفته باشد)، وقتی بسته درخواست DNS از یکی از نودهای شبکه به سرویس DNS می رسد، همانطور که د شکل ۴ نشان داده شده است، سیستم ارسال‌کننده درخواست در قسمت *IP Header* آدرس IP خود را قرار می‌دهد و به سمت سرویس DNS ارسال می کند. همچنین در لایه دوم مدل OSI یعنی لایه *DataLink* سیستم ارسال کننده درخواست DNS آدرس فیزیکی کارت شبکه خود را برای سرویس DNS ارسال می کند.

سیستم هایی که در زمینه شبکه مشغول به فعالیت هستند، دارای یک سطح درک متفاوتی از بسته های ارسالی درون شبکه هستند، اما نکته‌ای که وجود دارد این است که در صورتی که سیستم و یا تجهیزی سطح درکی تا یک لایه از مدل OSI را داشته باشد، بیانگر این موضوع است که لایه های پایین تر از آن را نیز درک می کند. باتوجه به این نکته، زمانی که سیستمی در خواست DNS خود را برای سرویس DNS ارسال می کند، سرویس DNS به دلیل آنکه سطح درک تا لایه هفتم مدل OSI یعنی لایه *Application* را دارد، لایه های زیرین را نیز درک می کند؛ از همین رو می تواند متوجه شود که سیستم

ارسال کننده بسته دارای چه آدرس فیزیکی می‌باشد.

با توجه به دو مورد بیان شده در پاراگراف های فوق، در این روش با استفاده از اطلاعات موجود در مسیریابی فعلی بسته و اطلاعاتی که از بسته های ارسال شده قبلی به دست می‌آید، می توان در ارتباط به مکان قرار گیری هر کدام از گره های موجود در شبکه، درک درستی را به سرویس DNS منتقل کرد. حال در صورتی که بسته ای به سرویس DNS برسد که متوجه شود آن بسته با آدرس IP و آدرس فیزیکی ارسالی با نودهای ثبت شده در مسیر ارسال همخوانی ندارد، مشکوک است و می‌بایست تصمیمی در این مورد اتخاذ شود. (شایان ذکر است که در روش های ارائه شده مبارزه با حملات DoS، مهم ترین اقدام، تشخیص درست حملات می‌باشد و فاز مقابله نیازمند صرف توان و هزینه کمتری می‌باشد. به عنوان مثال با توجه به تصمیم ادمین شبکه، می توان در این حالت میزان توان ارسال بسته از سمت نودی که آدرسش را عوض کرده به سمت سرویس DNS کم نمود تا میزان تأثر گذاری حمله تا حد زیادی کاهش یابد.)

کد منبع[1] به کار گیری شده در روش پیشنهادی به شرح زیر می‌باشد:

```
1. TABLE="/sbin/iptables"
2. WAN="eth0"
3. LAN="eth1"
4. $TABLE -F
5. $ TABLE -F INPUT
6. $ TABLE -F OUTPUT
7. $ TABLE -F FORWARD
8. $ TABLE -F -t mangle
9. $ TABLE -F -t nat
10. $ TABLE -X
11. $ TABLE -P INPUT DROP
12. $ TABLE -P OUTPUT ACCEPT
13. $ TABLE -P FORWARD ACCEPT
14. echo 1> /proc/sys/net/ipv4/ip_forward
15. $TABLE -t nat -A POSTROUTTING -O $WAN -j MASQUERADE
16. $TABLE -t nat -A PREROUTING -I $WAN -p tcp --dport 80 -j DNAT --to 10.0.0.1:80
17. $TABLE -A FORWARD -i $WAN -p tcp --dport 80 -m state --state NEW -j ACCEPT
18. $TABLE -A FORWARD -I $WAN -m state --state NEW,INVALID -j DROP
19. $TABLE -A INPUT -i lo -j ACCEPT
20. $TABLE -A INPUT -i $LAN -j ACCEPT
21. $TABLE -N Firewall
22. $TABLE -N Firewall -m limit --limit 10/minute -j LOG --log-prefix "Firewall: "
23. $TABLE -A Firewall -j DROP
24. $ TABLE -N Rejectwall
25. $ TABLE -A Rejecteall -m limit --limit 10/minute -j LOG –log-prefix "Rejectwall: "
26. $TABLE -A Rejecteall -j REJECT
27. $ TABLE -N Badflags
28. $ TABLE -A Badflags -m limit --limit 10/minute -j LOG --log-prefix "Badflags: "
```

1. Source Code

```
29. $ TABLE -A Badflags -j DROP
30. $ TABLE -A IPUT -p tcp --tcp-flags ACK,FIN FIN -j Badflags
31. $ TABLE -A IPUT -p tcp --tcp-flags ACK,PSH -j Badflags
32. $ TABLE -A IPUT -p tcp --tcp-flags ACK,URG URG -j Badflags
33. $ TABLE -A IPUT -p tcp --tcp-flags FIN,RST FIN,RST -j Badflags
34. $ TABLE -A IPUT -p tcp --tcp-flags SYN,FIN SYN,FIN -j Badflags
35. $ TABLE -A IPUT -p tcp --tcp-flags SYN,RST SIN,RST -j Badflags
36. $ TABLE -A IPUT -p tcp --tcp-flags ALL ALL -j Badflags
37. $ TABLE -A IPUT -p tcp --tcp-flags ALL NONE -j Badflags
38. $ TABLE -A IPUT -p tcp --tcp-flags ALL FIN,PSH,URG-j Badflags
39. $ TABLE -A IPUT -p tcp --tcp-flags SYN,FIN,PSH,URG -j Badflags
40. $ TABLE -A IPUT -p tcp --tcp-flags SYN,RST,ACK,FIN,URG -j Badflags
41. $TABLE -A INPUT -p icmp --icmp-type 0 -j ACCEPT
42. $TABLE -A INPUT -p icmp --icmp-type 3 -j ACCEPT
43. $TABLE -A INPUT -p icmp --icmp-type 11 -j ACCEPT
44. $TABLE -A INPUT -p icmp --icmp-type 8 -m limit –limit 1/second -j ACCEPT
45. $TABLE -A INPUT -p icmp -j Firewall
46. $TABLE -A INPUT -i $WAN -p tcp --drop 443 -j ACCEPT
47. $TABLE -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
48. $TABLE -A INPUT -p udp --sport 137 --dport 137 -j DROP
49. $TABLE -A INPUT -j Rejectwall
50. rate-limit {
51.slip 2;
52.Windows 15;
53.response-per-second 3;
54. refferrals-per-second 4;
55. nodata-per-second 6;
56. nxdomains-per-second 6;
57. errors-per-second 3;
58. all-per-second 30;
59. log-only yes;
 60. qps-scale 250;
61. ipv4-prefix-length 16;
62. ipv6-prefix-length 48;
63. };
64. }
```

در روش ارائه شده برای اینکه بتوانیم کلاینت‌ها و پاسخ هایی که برای آنها ارسال شده است را ردیابی و مدیریت کنیم، باید تمامی اطلاعاتی که مورد نیاز است درون حباب هایی به نام حباب‌های وضعیت[1] قرار گرفته شوند.

در این روش مقادیر به صورت چندگانه‌هایی[2] درون هر حباب قرار می گیرند. مقادیری که در این روش قرار می گیرند، به شرح زیر می‌باشند:

۱. زیردامنه سیستمی که قرار است پاسخی برای آن ایجاد و سپس ارسال شود.

۲. نام ورودی (همان نامی است که سرور به پاسخی که قرار است ایجاد شود اختصاص می دهد.)

1. State-Blobs
2. Tuple

۳. یک عبارت بولین که مشخص کننده روی دادن خطا می باشد.

نحوه اجرای این روش به همراه پردازش تولید یک پاسخ به شرح زیر می‌باشد:

۱. آدرس ارسال کننده درخواست به همراه یک نشانه[1] که بیانگر زیردامنه کلاینت می‌باشد، آماده سازی می‌شود.

(نشانه عبارتی است که به صورت یک عدد دورقمی و یا یک رقمی از عدد ۰ تا ۳۲ در آدرس IP نسخه چهار و از عدد ۰ تا ۱۲۸ برای آدرس IP نسخه ششم تعریف می‌شود و به صورت خلاصه بیانگر زیر شبکه مرتبط با آن آدرس IP است.)

۲. یک نام و یک عبارت بولین برای نشان دادن خطای احتمالی در نظر گرفته می‌شود.

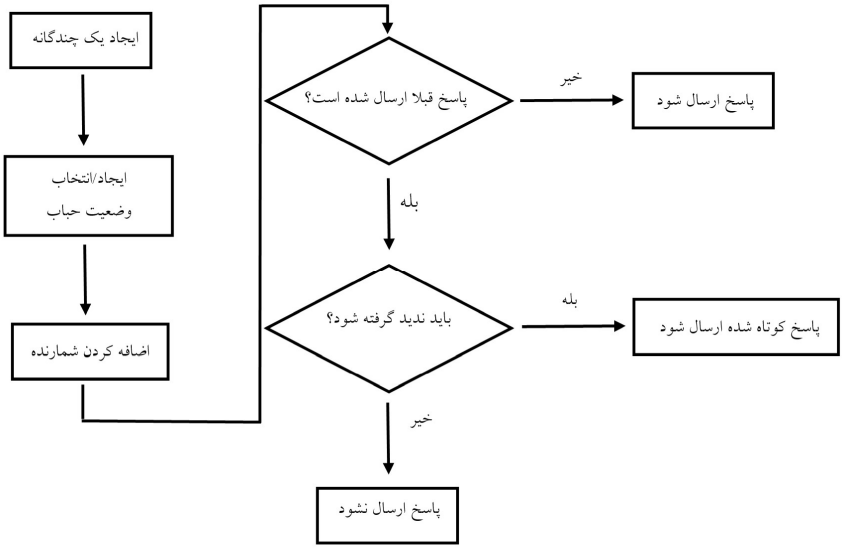
(درصورتی که مقدار RCODE برروی *FORMERR*، *REFUSED* یا *SERVFAIL* تنظیم شده باشد مقدار عبارت بولین «درست» می‌باشد.)

۳. مقدار چندگانه به صورت زیر تعریف می‌شود.

<وضعیت خطا, نام واردشده, آدرس IP>

حال سیستم برای هر کدام از چندگانه‌های ایجاد شده، محدودیت نرخ را بررسی می کند و درصورتی که مقدار مرتبط به هر حساب‌کاربری با مقداری که در سیستم تعریف شده است مغایرت کند، اجازه ساخت و ارسال پاسخ متناظر با آن درخواست را نمی‌دهد. روش انجام این عمل در شکل ۵ بیان شده است:

1. *Prefix*



شکل ۵ روش تصمیم‌گیری در ارتباط با تولید و یا عدم تولید پاسخ درخواست DNS

در ادامه به موجب درک بهتر این روش می‌بایست یکسری پارامترهایی که در این روش باید تنظیم شوند، معرفی شوند. در این روش باید ۱۰ پارامتر تعریف شده ذیل به درستی مقداردهی شوند تا مقدار احتمال تشخیص اشتباه حمله از طرف سرویس DNS کاهش یابد.

. RESPOSES-PER-SECOND:

این پارامتر نشان دهنده مقدار درخواست‌های یکسانی هستند که در ثانیه می توانند به سرور DNS برسند و مبداء ارسال کننده درخواست توسط سرویس DNS مسدود نشود. در صورتی که مقدار این پارامتر صفر باشد به این معنا است که ویژگی محدود کردن نرخ برای آن مبداء ارسال کننده غیرفعال می باشد و ارسال کننده پیام می واند به ارسال پیام ادامه دهد.

2. NXDOMAINS-PER-SECOND [۹]:

نشان دهنده تعداد درخواست‌هایی است که یک ارسال کننده پیام می تواند در یک ثانیه ارسال کند و پاسخ NXDOMAIN را در پی داشته باشد.

(زمانی سرویس DNS پاسخ NXDOMAIN را برمی گرداند که کلاینت اطلاعات یک گره در شبکه را درخواست کرده باشد که اصلا وجود خارجی نداشته باشد و یا رکورد متناظر آن در پایگاه داده سرویس DNS ثبت نشده باشد. از این روش حمله کنندگان استفاده می کنند و به صورت رندم عبارت هایی را

تحت قالب نام یک گره در شبکه ارسال می کنند و سرویس DNS مجبور به بررسی و پردازش این بسته‌ها می شود و پس از بررسی متوجه می‌شود که رکوردی متناظر با آن را ندارد و پاسخ *NXDOMAIN* را منتشر می کند.)

. *ERRORS-PER-SECOND*:

این پارامتر مشخص می کند که در ثانیه چه تعداد درخواست‌هایی می تواند به سرور برسد که پاسخ آن منجر به بروز خطا شود. منظور از خطا ایجاد پاسخ‌هایی با محتوای FORMERR ،REFUSED و یا *SERVFAIL* می باشد.

4. *WINDOW*:

یک بازه زمانی بر حسب ثانیه است که در آن مقدار نرخ ها مورد اندازه گیری قرار می گیرد.

برای مثال تصور کنید مقدار RESPONSES-PERSECOND در یک اندازه گیری ۵ باشد و مقدار *WINDOW* آن نیز برابر با ۵ باشد. در این حالت کلاینت محدود می‌شود که تنها در هر ثانیه می تواند ۵ درخواست یکسان و نمی تواند د ۵ عدد *WINDOW* بیشـ ۲۵ درخواست (۵*۵) برای سرویس DNS ارسال کند.

. *LEAK-RATE*:

این پارامتر بیانگر مقدار نرخ یک کلاینت در زمانی است که توسط سرویس DNS محدود شده است. به عنوان مثال د که مقدار این پارامتر ۳ باشد، بدین معنا است که در ازای هر سه بسته درخواستی که سرویس DNS دریافت می کند و از بین می برد، اجازه دارد یک بسته دریافت و پردازش شده و پاسخی برای آن ایجاد و برای قربانی ارسال شود.

6. *TC-RATE or SLIP*:

این پارامتر نیز همانند پارامتر قبلی می باشد، با این تفاوت که تعداد بسته هایی که اجازه پردازش و سپس ایجاد پاسخ را دارند، پاسخ‌هایی کوتاه شده ایجاد می شوند.

7. *MAX-TABLE-SIZE*:

بیشـ ن مقدار حبابی (پارامترهایی که اطلاعات کلاینت‌ها و بسته ها در آنها نگهداری می شود.) که یک سرور می تواند آنها را نگه‌دارد.

(پیشنهاد می‌شود این مقدار متناسب با مقدار *WINDOW* و با درنظر گرفتن بدترین زمانی که سرور می ند بسته های دریافتی را پردازش کند، تعیین شود. منظور از بدترین زمان، زمانی است که سرور در آن بازه زمانی با بیشترین مقدار دریافت بسته‌ها مواجه شود.)

8. *MIN-TABLE-SIZE*:

این پارامتر مقدار سایزی است که باید به یک جدول حباب اختصاص داده شود.

9. *IPv4-PREFIX-LENGTH*:

این پارامتر مشخص کننده طول هر آدرس *IP* نسخه ۴ است و بدین روش آدرس های *IP* نسخه ۴ را دسته بندی می کنیم.

(مقدار این پارامتر به صورت پیش فرض ۲۴ می باشد.)

10. *IPv6-PREFIX-LENGTH*:

این پارامتر مشخص کننده طول هر آدرس IP نسـ ۶ است و بدین روش آدرس های *IP* نسـ ه ۶ را دسته بندی می کنیم.

(مقدار این پارامتر به صورت پیش فرض ۵۶ می باشد.)

نکته‌ای که حائز اهمیت است این است که تمامی پارامترهای مطرح شده فوق اجباری نیستند که در تمام سرویس‌های DNS تنظیم شوند. حتی در بعضی از سرویس های DNS اصلا بعضی از پارامترها وجود ندارند که لازم باشد آنها را تنظیم کنیم. به عنوان مثال در سرویس DNS که برروی سیستم عامل لینوکس به نام BIND است اصلا پارامتری تحت عنوان *LEAK-RATE* وجود ندارد و به جای آن پارامتر دیگری تحت عنوان *ALL-PER-SECOND* وجود دارد که بیانگر بیشترین میزان تعداد تمامی پاسخ های تولید و ارسال شده توسط سرور DNS می باشد. در سرور DNS شرکت مایکروسافت نیز پارامتری تحت عنوان *NXDOMAIN-PER-SECOND* وجود ندارد و به جای آن دارای پارامتری به نام *MAXIMUM-PER-WINDOW* می‌باشد که همانند سرور BIND می باشد؛ با این تفاوت که برای تمامی پاسخ‌هایی است که در یک *WINDOW* ساخته شده است.

مقداردهی به هر کدام از متغیرهای بیان شده از اهمیت بسیار بالایی برخوردار بوده و نیازمند این است که با توجه به شرایط شبکه مقداردهی شوند. از طرفی این امکان وجود ندارد که یک مقدار دهی کلی برای تمامی شبکه ها معرفی شود.

برای اینکه مقداردهی پارامترهای بیان شده به درستی با توجه به شرایط شبکه تعیین شوند بهتر است این ویژگی را در حالت *LOGONLY* قرار دهیم. در این حالت به صورت حقیقی سرویس DNS بسته هایی که حمله تشخیص می‌دهد را از بین نمی برد و هیچ کلاینتی را نیز محدود نمی کند. بلکه در زمانی که ترافیکی را حمله تشخیص داد، گزارشی از آن در پایگاه داده ذخیره می کند تا ادمین با مطالعه آنها نحوه تشخیص سرویس را ارزیابی کند و با توجه به شرایط شبکه، سرویس را برای تشخیص هر چه درست تر حمله کمک کند. این کار را معمولا تا زمانی که سیستم شرایط مختلفی را تجربه کند، ادامه خواهند داد تا

سیستم بتواند به درستی حمله را تشخیص دهد.

راهکارهایی برای انتخاب و محاسبه مقدار پارامترهای بیان شده وجود دارد که در این قسمت به بعضی از آنها می پردازیم:

۱. پارامتر *Responses Per-Second*:

برای محاسبه این پارامتر بهتر است که طی بازه زمانی مشخصی ترافیکی که وارد سرویس DNS و از آن خارج می شود مانیتور شده و لاگ‌های تولید شده در حالت *LOGONLY* مورد بررسی قرار گیرد تا مشخص شود که سرویس DNS بصورت میانگین بیشتر به چه زیر شبکه هایی خدمات می‌دهد. (البته شایان ذکر است که برای این کار باید از میان لاگ های تولید شده برروی لاگ‌هایی که مربوط به بزرگ ترین بسته های سرویس DNS است، تمرکز شود.) به عنوان مثال باید مشخص شود که بزرگ ترین بسته های تولیدشده سرویس DNS مربوط به زیر شبکه‌هایی از جنس آدرس های *IP* نسخه ۴ با ۲۴/ هستند یا از جنس آدرس های *IP* نسخه ۶ با ۵۶/ هستند.

اما سوالی که اینجا مطرح می شود این است که چگونه اندازه بیشترین تعداد بسته‌های پاسخ را برای یک زیر شبکه خاص محاسبه کرد؟ برای این کار می‌بایست از فرمول زیر استفاده شود:

$$\text{تعداد بیشترین پاسخ ها} \approx \frac{\text{بیشترین میزان پهنای باند}}{\text{اندازه بزرگترین پاسخ صادرشده}}$$

۲. اندازه *WINDOW*:

این پارامتر همانند یک مجازات برای کلاینت‌ها و گره‌هایی است که ترافیک های مشکوک و یا مخرب ارسال می‌کنند. این بدین معنا است که هر چه مقدار این پارامتر بیشـ باشد، کلاینت دیرتر می تواند پاسخ‌هایی نرمال از سرویس DNS دریافت کند و بسته‌های درخواست خودش را به سرویس DNS برای پردازش ارسال کند. هرچه مقدار پارامتر RESPONSES-PER-SECOND کوچک‌تر باشد و مقدار *WINDOWS* بیشتر باشد، سرویس DNS ایمن‌تر خواهد بود.

۳. طول نشانه‌های مرتبط به *IPv4* و *IPv6*[1]:

با مقداردهی و تعریف این پارامتر می توان کلاینت‌هایی که از سرویس DNS خدمات می گیرند را براساس زیرشبکه‌هایشان به گروه مجزا تقسیم بندی کرد تا بدین وسیله مسیر ارتباط با DNS ایمن‌تر شود. منظور از ایمن سازی مسیر ارتباط با DNS رمزنگاری و یا ایمن‌سازی بسته های ارسالی نمی‌باشد، بلکه منظور این است که سرویس DNS به دلیل اینکه این مسیرهای ارتباطی را می ناسد متوجه می‌شود که این بسته در حقیقت از گره‌ای که آدرسش در قسمت آدرس مبداء بسته قرار دارد، ارسال شده است یا خیر.

1. *IPv4 and IPv6 Prefix Length*

مقدار این پارامتر برای آدرس‌های IP نسخه چهارم به صورت پیش فرض ۲۴ و برای آدرس‌های نسخه ششم مقدار ۵۶ می‌باشد. حال در صورتی که این مقادیر را کاهش دهیم باعث می‌شود که اندازه هر زیرشبکه کاهش و تعداد گروه‌ها افزایش یابد؛ این امکان می‌تواند موجب شود که کاربران قانونی با کاربرانی که به عنوان حمله کننده تشخیص داده شدند در یک گروه قرار بگیرند و سیستم در شناسایی بسته‌های ارسالی از طرف آنان دچار تشخیص اشتباه شود. در مقابل هرچه مقدار این پارامتر بیشتر باشد، شانس این مورد کاهش مییابد.

۴. پارامتر SLIP:

مقدار این پارامتر موازنه‌ای میان تعداد بسته‌های پاسخی که می توانند ارسال شوند و میزان تشخیص اشتباه برقرار می کند. حال هر چه این میزان بیشتر باشد باعث می‌شود مقدار کمتری بسته پاسخ بتواند به سمت آدرس مبداء ایجاد و ارسال شود و در نتیجه میزان تشخیص اشتباه حمله بیشتر شده و در نهایت منجر شود که بسته‌های پاسخی که می‌بایست در نتیجه دریافت درخواستی که از سوی کلاینت صادر شده است، ایجاد شوند، تشکیل نشوند.

۴. مطالعه کاربردی

نتیجه تست صحت عملکرد در ارتباط با روش ارائه شده در پنج قسمت به شرح زیر به صورت تلخیص شده آورده شده است:

۱. تشخیص درخواست‌هایی با زیر دامنه‌هایی که وجود ندارند

همان طور که پیش تر هم گفته شد، توصیه می‌شود از این ویژگی برای سرورهایی که دارای رکورد ناحیه می‌باشند، استفاده شود. اما چرا؟

در شکل ۶ پاسخ هایی که سرور بالاترین سطح دامنه[1]، به سمت حمله کننده ارسال کرده نمایش داده شده است.

```
... (hclgtixc.67.tld): query: hclgtixc.67.tld IN A +
... (piruzcoa.67.tld): query: piruzcoa.67.tld IN A +
... (ndhpquof.67.tld): query: ndhpquof.67.tld IN A +
... (klmgudbx.67.tld): query: klmgudbx.67.tld IN A +
... (stfvcqle.67.tld): query: stfvcqle.67.tld IN A +
```

شکل ۶ نمونه پاسخ های سرور DNS در زمان حمله

همان طور که مشاهده می کنید درخواست هایی با دامنه های یکسان اما با زیردامنه هایی متفاوت توسط سرور DNS دریافت شده‌است. در شکل ۷ گزارشی از وضعیت و عملکرد محدود سازی پاسخ های ارسال شده نمایش داده شده است. همان طور که مشاهده می کنید به دلیل این که دامنه‌های تمامی درخواست های ارسالی توسط حمله کننده یکسان می‌باشد، سیستم تمام درخواست‌هایی که با آن دامنه

1. *Top Level Domain (TLD)*

ارسال شده است را حمله تشخیص داده و باعث می شود تمامی درخواست‌های دریافتی را از بین ببرد و دیگر پاسخی برای آنها ایجاد نکند.

```
root@ns1:~# tail -n 5 /var/log/named/rate-limit.log
... (hclgtixc.67.tld): would rate limit drop referral ... for 67.tld IN
... (piruzcoa.67.tld): would rate limit slip referral ... for 67.tld IN
... (ndhpquof.67.tld): would rate limit drop referral ... for 67.tld IN
... (klmgudbx.67.tld): would rate limit slip referral ... for 67.tld IN
... (stfvcqle.67.tld): would rate limit drop referral ... for 67.tld IN
```

شکل ۷ نمونه پاسخ های سرور DNS در بله به حمله صورت گرفته

در صورتی که اگر این ویژگی برروی سرورهای بازگشتی مورد استفاده قرار می گرفت، به دلیل اینکه درخواست‌هایی که دریافت می کنند را برای سرورهای بالاترین سطح دامنه ارسال می کنند دیگر نمی توانند یکسان بودن دامنه را تشخیص دهند. از طرف دیگر این ویژگی را نیز نمی توان برروی سایر سرورهای DNS که مالک و ایجاد کننده آن ناحیه نیستند، ایجاد نمود زیرا آنها نیز می‌بایست برای استعلام، درخواست را برای سرورهای بالاترین سطح دامنه ارسال کنند و این ترافیک حمله دوباره در شبکه پخش خواهد شد و موجب می‌شود پهنای باند شبکه را به خود اختصاص دهد.

۲. تشخیص درخواست‌هایی با دامنه‌هایی که وجود ندارند

در این حالت نیز به دلیل آنکه سرورهایی که در بالاترین سطح دامنه قرار دارند لیست تمام دامنه‌ها را در اختیار دارند می توانند به راحتی درخواست‌هایی که از یک کلاینت برای دامنه‌های متفاوت ارسال می شود را تشخیص دهند و تمامی آنها را از بین ببرند.

همان طور که در شکل ۸ مشاهده می کنید درخواست‌هایی برای سرور DNS با دامنه هایی متفاوت ارسال شده است و به راحتی سرور DNS تشخیص داده است که این درخواست‌ها از جانب یک هکر صادر شده و تمامی آنها را از بین برده است و یا بنا به تنظیماتی که ادمین سرور انجام داده، بعضی از درخواست‌های دریافت شده را SLIP کرده است.

```
root@ns1:~# tail -n 10 /var/log/named/queries.log
... (ephpsffzgb.205851.tld): query: ephpsffzgb.205851.tld IN A +
... (sorzqloepv.256681.tld): query: sorzqloepv.256681.tld IN NS +
... (niqvmmygdo.239178.tld): query: niqvmmygdo.239178.tld IN NS +
... (sguupwjsnr.294212.tld): query: sguupwjsnr.294212.tld IN AAAA +
... (oxofpeeskj.256061.tld): query: oxofpeeskj.256061.tld IN AAAA +
root@ns1:~# tail -n 10 /var/log/named/rate-limit.log
... (ephpsffzgb.205851.tld): would rate limit drop NXDOMAIN ... for tld
... (sorzqloepv.256681.tld): would rate limit slip NXDOMAIN ... for tld
... (niqvmmygdo.239178.tld): would rate limit drop NXDOMAIN ... for tld
... (sguupwjsnr.294212.tld): would rate limit slip NXDOMAIN ... for tld
... (oxofpeeskj.256061.tld): would rate limit drop NXDOMAIN ... for tld
```

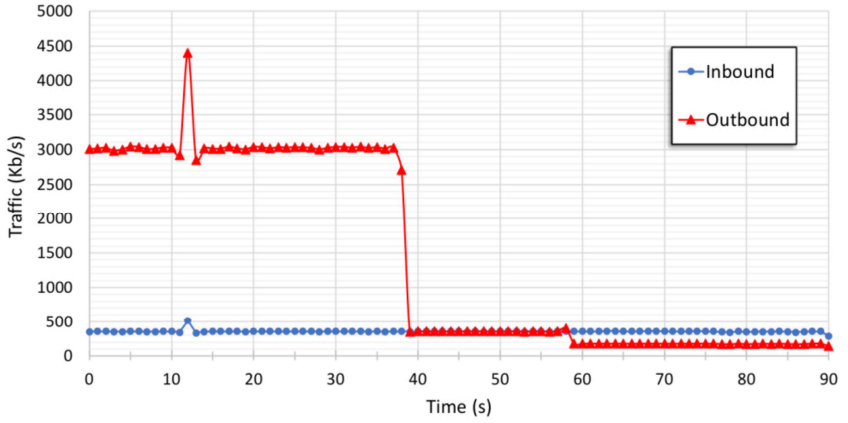
شکل ۸ واکنش سرور DNS به درخواست هایی که حمله کننده با دا نه هایی متفاوت ایجاد کرده است.

این موضوع که سرویس DNS درخواست‌هایی که برای دامنه هایی که اصلا وجودخارجی ندارند صادر می‌شوند را در سطوح بالا چگونه تشخیص می‌دهد نیز بدین صورت است که سرور DNS درخواست‌ها را دریافت و سپس برای تولید پاسخ اقدام می‌کند و به پاسخ NXDOMAIN که هنگامی که دامنه‌ای وجود خارجی ندارد ایجاد می شود می رسد، از طرفی دیگر درخواست‌ها به صورت مکرر به سرور DNS می رسد و از کنار هم قرار دادن اینگونه درخواست‌ها سرور نتیجه می گیرد که تمامی این درخواست‌ها از طرف یک هکر می‌باشد و نمی بایست پاسخی را برای آنها ایجاد و ارسال کند.

همانطور که بیان شد در این گونه از حملات، میزان درصد نام یی که توسط سرور DNS پاسخ داده شده اند، صفر درصد می‌باشد. نمودار ۱ و جدول ۲ نمایانگر محدود سازی پاسخ DNS در مقابله با حملاتی است که دامنه‌هایی را درخواست می‌دهند که اصلا وجود خارجی ندارند.

جدول ۲: تاثیر راهکار ارائه شده در ز ی که حمله کننده درخوا تهایی با دامنه هایی که وجود خارجی ندارند را ارسال می کند.

مقدار SLIP	مقدار احتمال تشخیص اشتباه	مقدار ترافیک ورودی	مقدار ترافیک خروجی
SLIP1	%۰	۳۴۷ کیلوبیت بر ثانیه	۳۴۷ کیلوبیت بر ثانیه
SLIP2	%۵۰	۳۴۷ کیلوبیت بر ثانیه	۱۷۴ کیلوبیت بر ثانیه
SLIP3	%۶۶.۶	۳۴۸ کیلوبیت بر ثانیه	۱۱۵ کیلوبیت بر ثانیه
SLIP5	%۸۰	۳۴۷ کیلوبیت بر ثانیه	۶۹ کیلوبیت بر ثانیه
SLIP10	%۹۰	۳۴۸ کیلوبیت بر ثانیه	۳۵ کیلوبیت بر ثانیه

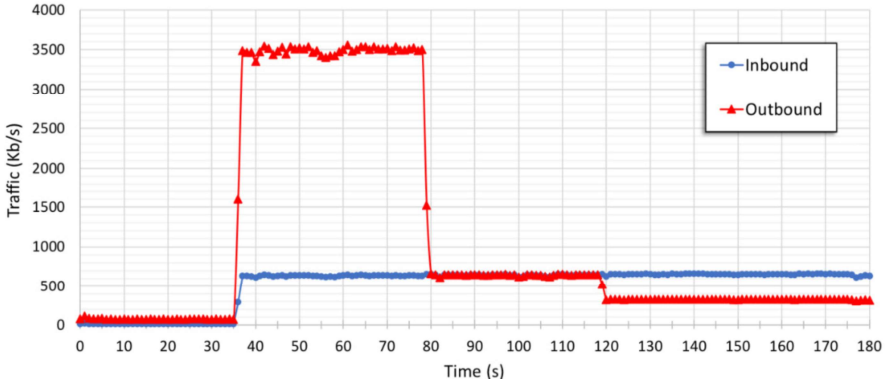


نمودار ۱: پاسخ سرور در زمان حمله با دا نه هایی که وجود خارجی ندار د (با استفاده از راهکار ارائه شده).

در این آزمایش در زمان ۳۸ پاسخ‌ها محدودسازی شدند؛ آن هم با مقدار SLIP برابر یک و تا زمان ۹۰ با مقدار SLIP برابر با دو، آزمایش ادامه پیدا کرده است.

۳. تشخیص درخواست‌های تکراری با پاسخ هایی با طول زیاد

یکی از انواع حملات DOS که به صورت گسترده در دنیا سرورهای DNS را تحت تاثیر خود قرار می‌دهند حملاتی هستند که هکر بسته‌های درخواست تکراری را با نرخ بالا برای سرور DNS ارسال می کند. همان طور که پیش تر نیز گفته شد، در این گونه حملات حمله کننده با ارسال درخواست‌های «ANY» باعث می شود که سرور DNS با دریافت اینگونه درخواست‌ها تمامی اطلاعاتی که مرتبط با آن رکورد در اختیار دارد را برای یک مقصدی خاص ارسال کند و درواقع حمله کننده از این روش برای حملات تقویت شده استفاده می کند. برای نمونه نمود ر ۲ نمایانگر نرخ ورودی و خروجی یک سرور DNS می‌باشد که تحت اینگونه حملات قرار گرفته است و همان طور که مشخص است نرخ ورودی آن به صورت میانگین ۶۰۰ کیلوبیت برثانیه است درحالی که چون از درخواست های «ANY» استفاده کرده است، میزان نرخ خروجی آن سرور حدود ۳/۵ مگابیت بر ثانیه شده است.



نمودار ۲: سرور در زمان حمله با درخواست‌های ANY توسط حمله کننده صادر شده ا ت (با استفاده از راهکار ارائه شده).

در این تست مقدار SLIP برابر با یک قرار داده شده است و حدود ۱۰ پاسخ در ثانیه برای درخواست‌هایی که دریافت می‌شود، تولید و به سمت مقصد ارسال می‌شود. در این آزمایش به نوبت میزان SLIP افزایش داده شده‌است که به صورت جدول ۳، میزان تاثیر افزایش SLIP در برابر حملات درخواست‌های تکراری نمایش داده شده است.

شایان ذکر است در این تست ویژگی SLIP به همراه کوتاه کردن پیام باهم فعال شده است تا پس از آنکه ترافیک دریافتی توسط سرور DNS حمله تشخیص داده شد، تمامی بسته های درخواست دریافتی

از بین نروند و احتمال تشخیص اشتباه توسط سرور کاهش یابد و به همین دلیل کاربران قانونی بتوانند به فعالیت خود ادامه دهند.

در این آزمایش از ثانیه صفر تا ۸۰ سیستم دارای مقدار SLIP برابر با یک می‌باشد و به مدت ۱۰۰ ثانیه دارای SLIP برابر ۲ می‌باشد.

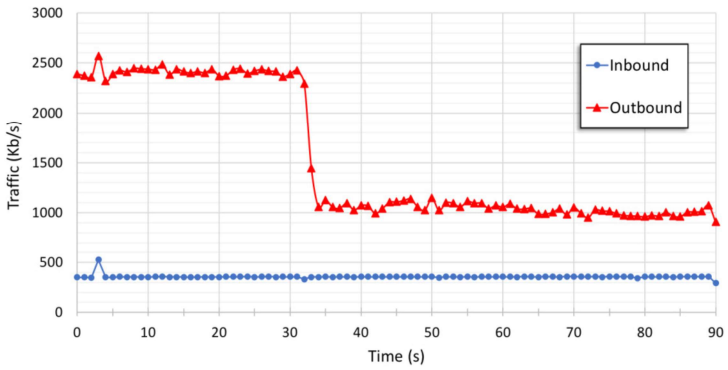
جدول ۳: پاسخ سرور در زمان حمله با درخواست های تکراری و نتیجه افزا مقدار SLIP (با استفاده از راهکار ارائه شده).

مقدار SLIP	مقدار احتمال تشخیص اشتباه	مقدار ترافیک ورودی	مقدار ترافیک خروجی
SLIP1	%۰	۶۱۸ کیلوبیت بر ثانیه	۶۲۵ کیلوبیت بر ثانیه
SLIP2	%۵۰	۶۲۵ کیلوبیت بر ثانیه	۳۱۲ کیلوبیت بر ثانیه
SLIP3	%۶۶.۶	۶۳۰ کیلوبیت بر ثانیه	۲۱۰ کیلوبیت بر ثانیه
SLIP5	%۸۰	۶۳۵ کیلوبیت بر ثانیه	۱۲۶ کیلوبیت بر ثانیه
SLIP10	%۹۰	۶۴۰ کیلوبیت بر ثانیه	کیلوبیت بر ثانیه

٤. تشخیص حملات درخواست های ترکیبی

در این نوع حملات، حمله کننده میزان پیچیدگی حملات خود را افزایش داده و از درخواست هایی ترکیبی استفاده می کند. منظور از درخواست‌های ترکیبی این است که حدود نیمی از درخواست ها را به گونه ای ارسال می کند که دارای پاسخ NXDOMAIN باشد و یا به عبارت دیگر دامنه‌هایی را مورد درخواست قرار می دهد که عملا یا وجود خارجی ندارند و یا این رکورد در سرویس DNS ثبت نشده باشد؛ مابقی درخواست ها نیز به وسیله اطلاعاتی که حمله کننده از قبل نسبت به سرویس DNS که مورد حمله قرار داده است تجمیع کرده است، درخواست‌هایی هستند که رکورد آنها در سرویس DNS وجود دارد و باعث می شود که سرویس DNS پاسخی را آماده کرده و برای حمله کننده ارسال کند.

همانطور که توضیح داده شد در این حملات به دلیل اینکه حمله کننده در کنار رفتار غیرعادی خود، رفتار عادی نیز دارد باعث می‌شود که تشخیص این نوع حملات و مقابله با آنها برای تیم امنیت سخت تر شود. در این نوع از حملات همانطور که در نمودار ۳ و جدول ۴ نمایش داده شده است، بدلیل آنکه رفتار غیرعادی حمله کننده با رفتار عادی ترکیب شده و میزان رفتار غیر عادی آن بگونه‌ای نیست که موجب شود سرویس DNS شروع به محدود کردن پاسخ‌ها بکند، حمله کننده موفق تر است و می تواند تا حد قابل توجهی میزان ترافیک و بارپردازشی زیادی را بر سرور تحمیل کند.



نمودار ۳: پاسخ سرور در زمان حملات ترکیبی (با استفاده از راهکار ارائه شده).

همان طور که در نمود خص است تا حدی از میزان حجم بسته‌های خروجی کاسته شده است، اما بازهم بارپردازشی و ترافیکی قابل توجه به سرویس تحمیل شده است. در این آزمایش تا ثانیه سی ام میزان پارامتر SLIP برابر با یک بوده است و از آن پس میزان SLIP برابر با دو شده است.

جدول ۴: سخ سرور در زمان حمله به حملات ترکیبی و تاثیر افزایش SLIP (با استفاده از راهکار ارائه شده).

مقدار SLIP	مقدار احتمال تشخیص اشتباه	مقدار ترافیک ورودی	مقدار ترافیک خروجی
SLIP1	٪۰	۳۴۷ کیلوبیت بر ثانیه	۱ مگابایت بر ثانیه
SLIP2	٪۵۰	۳۴۸ کیلوبیت بر ثانیه	۹۴۰ کیلوبیت بر ثانیه
SLIP3	٪۶۶.۶	۳۴۷ کیلوبیت بر ثانیه	۹۲۰ کیلوبیت بر ثانیه
SLIP5	٪۸۰	۳۴۷ کیلوبیت بر ثانیه	۸۹۲ کیلوبیت بر ثانیه
SLIP10	٪۹۰	۳۴۷ کیلوبیت بر ثانیه	۸۸۱ کیلوبیت بر ثانیه

همان طور که در جدول ۴ مشخص شده است، در روش ارائه شده با افزایش مقدار پارامتر SLIP از میزان ترافیک خروجی سرویس DNS حدقابل توجهی کاسته نشده و بدین ترتیب در این نوع از حملات روش ارائه شده دارای اثر گذاری کمی می‌باشد.

۵. تشخیص حملات ناحیه‌های شاخص گذاری شده

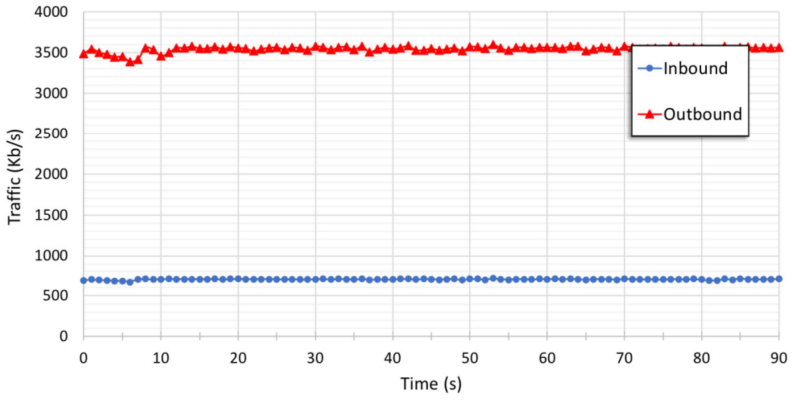
در این حالت حمله کننده همانند یک گره قانونی در شبکه عمل می‌کند. به عبارت دیگر در این نوع از حملات حمله کننده می‌بایست قبل از اجرایی کردن حمله خود اطلاعات بسیار زیادی مرتبط با رکوردهای موجود در سرور DNS کسب کند تا از این طریق بتواند درخواست‌هایی ارسال کند که با پاسخ

NXDOMAIN مواجه نشود. در این حالت تمامی درخواست‌هایی که ارسال می‌کند، رکوردهای متناظر آن در سرور DNS موجود می باشد و هیچ پیغام خطایی دریافت نمی کند.

از طرفی حمله کننده نیز می ست این اطلاعات را نیز بدست بیاورد که تنظیمات مرتبط با محدود سازی پاسخ در سرویس DNS به چه صورت انجام شده است که نرخ درخواست های قانونی که ارسال می کند از نرخ تنظیم شده در سرویس DNS بیشتر نباشد تا منجر به محدود سازی درخواست هایش شود. همچنین حمله کننده می‌تواند در صورتی که نرخ تعریف شده در سرویس DNS پایین باشد، از شبکه ای از ربات ها استفاده کند و درخواست هایش را از طریق آنها به سمت سرویس قربانی ارسال کند و بار پردازشی و ترافیکی بالایی را به سرور قربانی تحمیل کند تا بدین صورت دیگر درخواست هایش با محدود شدن پاسخ و مسدودسازی دریافت درخواست ها مواجه نشود.

همان طور که بیان شد در این روش حمله کننده درست همانند یک نود قانونی عمل می د تنها با این تفاوت که تعداد درخواست هایی که به سمت سیستم قربانی ارسال می کند دارای نرخ بسیار بالایی است که این روش را نیز می واند بابه خدمت گرفتن تعداد زیادی شبکه ربات ها مرتفع سازد.

در این حالت دیگر روش پیشنهادی هیچ گونه کارایی ندارد و به ازای تمامی درخواست ها پاسخ های متناسب با آنها تولید می‌شود. در نمودار ۴ و جدول ۵ حاصل نتیجه اینگونه حملات مشخص شده است.



نمودار ۴: پاسخ سرور در زمان حمله با استفاده از ناحیه‌های شاخص گذاری شده (با استفاده از راهکار ارائه شده).

در آزمایش فوق از ثانیه صفر تا ثانیه ۳۰ مقدار پارامتر SLIP برابر با مقدار یک و از آن پس مقدار پارامتر SLIP برابر با دو می‌باشد.

جدول ۵: پاسخ سرور در زمان حمله با استفاده از ناحیه‌های شاخص گذاری شده و تاثیر افزایش مقدار SLIP (با استفاده از راهکار ارائه شده).

مقدار SLIP	مقدار احتمال تشخیص اشتباه	مقدار ترافیک ورودی	مقدار ترافیک خروجی
SLIP1	٪۰	۶۳۰ کیلوبیت بر ثانیه	۲،۴۵ مگابیت بر ثانیه
SLIP2	٪۵۰	۶۳۰ کیلوبیت بر ثانیه	۲،۴ مگابیت بر ثانیه
SLIP3	٪۶۶،۶	۶۳۰ کیلوبیت بر ثانیه	۲،۳۸ مگابیت بر ثانیه
SLIP5	٪۸۰	۶۳۱ کیلوبیت بر ثانیه	۲،۳۶ مگابیت بر ثانیه
SLIP10	٪۹۰	۶۳۰ کیلوبیت بر ثانیه	۲،۳۴ مگابیت بر ثانیه

۵. نتیجه گیری

همان طور که در نمودار ۴ و جدول مشخص شده است، تغییر در مقدار پارامتر SLIP هیچگونه تاثیری در مقابله با حمله و تخفیف در اثرات حمله با استفاده از ناحیه های شاخص گذاری شده ندارد. دلیل این امر هم این است که اصلا این نوع از حملات توسط این ویژگی قابل شناسایی نمی‌باشند که بخواهد با آن مقابله شود. به عبارت دیگر نقص این روش ابتدا در تشخیص این نوع از حملات می‌باشد.

همانطور که در گزارشات و نمودارهای ارائه شده در این مقاله مشخص شده است، روش ارائه شده برای حملات بسیار پیچیده که از ترکیب کردن حالات فوق استفاده می کنند دارای تاثیر گذاری کمتری می باشد؛ زیرا تشخیص این نوع حملات بسیار سخت می‌باشد و در صورت اینکه حساسیت این سامانه برای بررسی و آنالیز ترافیک‌های دریافتی توسط ادمین شبکه افزایش یابد احتمال تشخیص ترافیک قانونی کلاینت‌ها به عنوان ترافیک حمله (تشخیص اشتباه) بسیار بیشه می شود و همین امر سبب می شود تا سیستم با اختلال مواجه بشود.

در ارتباط با حملاتی که از ناحیه‌های شاخص گذاری شده استفاده می کنند می بایست گفت که به علت آنکه حمله کننده باید اطلاعات زیادی از سیاست‌های امنیتی و حفاظتی سرویس DNS، سرور DNS و همچنین اطلاعات بسیار زیادی از رکوردها و دامنه‌های ثبت شده در سرویس DNS داشته باشد، روش پیشنهادی بهره چندانی ندارد و ترافیک و درخواست‌های دریافتی را پردازش کرده و پاسخ متناسب با آن را برای کلاینت ارسال می کند.

منابع

[۱] *Aljuhani, A.* (2021)**. *Machine learning approaches for combating distributed denial of service attacks in modern networking environments.*** *IEEE Access*

[۲] *Deccio, C., Hilton, A., Briggs, M., Avery, T., & Richardson, R.* (202 **. *Behind closed doors: A network tale of spoofing, intrusion, and false dns security.*** *Proceedings of the ACM Internet Measurement Conference,*

[۳] *Dooley, M. and T. Rooney, DNS Security Management.* 2017: *Wiley.*

[۴] *Fidele, K. A., & Hartanto, A.* (2019)**. *DoS Attack Prevention Using Rule-Based Sniffing Technique and Firewall in Cloud Computing.*** *E3S Web of Conferences*

[۵] *Ghorpade, M.* (2018)**. *Constructing Inter Domain Packet Filter for Controlling IP Spoofing.***

[۶] *Gupta, V., & Sharma, E.* (2018)**. *Mitigating DNS amplification attacks using a set of geographically distributed SDN routers.*** 2018 *International Conference on Advances in Computing, Communications and Informatics (ICACCI),*

[۷] *Harikrishnan, V., Sanket, H., Sahazeer, K., Vinay, S., & Honnavalli, P. B.* (2022) ***Mitigation of DDoS attacks using honeypot and firewall.*** *In Proceedings of Data Analytics and Management* (pp. 625-635). *Springer.*

[۸] *Hasegawa, K., Kondo, D., & Tode, H.* (2021)**. *FQDN-based whitelist filter on a DNS cache server against the DNS water torture attack.*** 2021 *IFIP/IEEE International Symposium on Integrated Network Management (IM),*

[۹] *Hasegawa, K., Kondo, D., & Tode, H.* (2021)**. *FQDN-based whitelist filter on a DNS cache server against the DNS water torture attack.*** 2021 *IFIP/IEEE International Symposium on Integrated Network Management (IM),*

[۱۰] *Hasegawa, K., Kondo, D., & Tode, H.* (2021)**. *FQDN-based whitelist filter on a DNS cache server against the DNS water torture attack.*** 2021 *IFIP/IEEE International Symposium on Integrated Network Management (IM),*

[۱۱] *Ismail, S., Hassen, H. R., Just, M., & Zantout, H.* (2021)**. *A review of amplification-based distributed denial of service attacks and their mitigation.*** *Computers & Security,* 109, 102380.

[۱۲] *Jadin, M., et al., Securing multipath TCP: Design & implementation, in IEEE INFOCOM 2017 - IEEE Conference on Computer Communications.* 2017, *IEEE.*

[۱۳] Kambourakis, G., et al. (2007). Kambourakis, G., et al., ***A Fair Solution to DNS Amplification Attacks, in Second International Workshop on Digital*** Forensics and Incident Analysis (WDFIA 2007). 2007, IEEE

[۱۴] Kshirsagar, D., & Kumar, S. (2022 **. *A feature reduction based reflected and exploited DDoS attacks detection system.*** Journal of Ambient Intelligence and Humanized Computing, 13(1), 393-۴۰۵.

[۱۵] Li, M., Li, Q., Xuan, G., & Guo, D. (2021)**. *Identifying compromised hosts under APT using DNS request sequences.*** Journal of Parallel and Distributed Computing, 152, 67-78.

[۱۶] Liu, Z., X. Yin, and H. Lee. (2014)**. Liu, Z.**, X. Yin, and H. Lee. An efficient defense scheme against sip dos attack in sdn using cloud sfw. in 2014 Ninth Asia Joint Conference on Information Security. 2014. IEEE.

[۱۷] Lu, K., Chai, T., Xu, H., Prasad, S., Yan, J., & Zhang, Z. (2021)**. Research on Unexpected DNS Response from Open DNS Resolvers.** The Computer Journal.

[۱۸] Moura, G. C., Castro, S., Heidemann, J., & Hardaker, W. (2021)**. *tsuNAME: exploiting misconfiguration and vulnerability to DDoS DNS.*** Proceedings of the 21 st ACM Internet Measurement Conference,

[۱۹] Myneni, S., Chowdhary, A., Huang, D., & Alshamrani, A. (2022)**. *SmartDefense: A distributed deep defense against DDoS attacks with edge computing.*** Computer Networks, 209, 108874.

[۲۰] Santoso, D., Noertjahyana, A., & Andjarwirawan, J. (2022)**. *Implementasi dan Analisa Snort dan Suricata Sebagai IDS dan IPS Untuk Mencegah Serangan DOS dan DDOS.*** Jurnal Infra, 10(1), 142-۱۴۷.

[۲۱] Segawa, S., Masuda, H., & Mori, M. (2019)**. *Proposal and Prototype of DNS Server Firewall with Flexible Response Control Mechanism.*** 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD),

[۲۲] Tripathi, N., & Hubballi, N. (2021)**. *Application layer denial-of-service attacks and defense mechanisms: a survey.*** ACM Computing Surveys (CSUR).