مقابله با حمله های "از کار اندازی سرویس" در شبکه های کامپیوتری

احمدرضا شرافت

دانشیار بخش مهندسی برق – دانشکده فنی – دانشگاه تربیت مدرس

مهران سلیمان فلاح

فارغ التحصیل دکتری بخش مهندسی برق – دانشکده فنی – دانشگاه تربیت مدرس

(تاریخ دریافت ۸۰/۵/۲٤، تاریخ تصویب ۸۱/٤/۱۵)

چکیده

قابلیت دسترسی به معنای ارائه سرویس های مورد انتظار و تعریف شده سیستم در زمانهای مورد نظر به کاربران است. حمله هایی که با هدف تهدید یا از بین بردن قابلیت دسترسی انجام می شوند، حمله های از کار اندازی سرویس نامیده می شوند. در این مقاله دسته بندی جدیدی از این حمله ها را ارائه کرده ایم. سپس بر اساس دسته بندی ارائه شده برخی اصول لازم برای طراحی پروتکل هایی که در مقابل این نوع حمله ها مقاوم باشند را مطرح کرده و آنها را تحلیل کرده ایم. در این مقاله همچنین نشان داده ایم که برخی از پروتکل های موجود در TCP/IP با اصول مطرح شده سازگار نیستند و به این دلیل در مقابل حمله های از کار اندازی سرویس آسیب پذیرند. همچنین در این مقاله برخی از حمله های از کار اندازی را بررسی کرده و راه حل هایی برای سازگار کردن پروتکل ها با اصول مطرح شده ارائه کرده ایم.

واژه های کلیدی: از کار اندازی سرویس، قابلیت دسترسی، امنیت، شبکه های کامپیوتری، TCP/IP.

مقدمه

هر سیستم امن اطلاعات باید دارای سه ویژگی باشد: از دستیابی غیر مجاز به اطلاعات جلوگیری کند (محرمانگی)، تلاشهای غیر مجاز برای تغییر اطلاعات را رد کند (صحت) و دسترسی به سرویس ها را تضمین کند (قابلیت دسترسی) [۱]. قابلیت دسترسی به معنای امکان دستیابی به سرویس های تعریف شده در درون سیستم توسط کاربران در زمانهای مورد انتظار است.

حمله هایی که قابلیت دسترسی را تهدید می کنند

حمله های از کار اندازی سرویس[۱] یا به اختصار DoS نامیده می شوند. در حمله های DoS حمله کننده با به کار گیری سرویس های ارائه شده توسط سیستم حمله خود را پی ریزی می کند. پس از اینکه یک سیستم دچار حمله های DoS شد برخی سرویسهای آن از کار می افتند و همچنین ممکن است کلیه فعالیتهای سیستم متوقف شود. یک نمونه از این حمله ها با نام حمله Smurfing در فوریه سال ۲۰۰۰ برای کامپیوترهای میزبان Yahoo رخ داد که در آن به مدت ٤ ساعت سیستم تحت حمله بود و در این فاصله زمانی درخواست میلیونها کاربر برای سرویس های پست الکترونیکی و فهرستهای زمانی رد می شد [۲]. همچنین گزارشها نشان می دهند روزانه تعداد زیادی از سرویس دهنده ها دچار چنین حمله هایی می شوند. این حمله ها باعث قطع شدن ارتباط کاربران با سرویس دهنده در هنگام استفاده از سرویس ها، تلاشی[۲] سیستم، مجددا راه اندازی شدن[۳] سیستم و همچنین ارائه سرویس های متفاوت با آنچه کاربران درخواست کرده اند شده است[۳،٤،٥،٦،۷،۸،۹].

حمله های از کار اندازی سرویس با روشهای مختلفی انجام می شوند. یکی از روشها اشباع منابع محدود در یک سرویس دهنده است. در این نوع از حمله های از کار اندازی سرویس، حمله کننده با استفاده تکراری از منابع محدود سیستم به گونه ای که سرویس دهنده زمان کافی برای آزاد کردن این منابع را نداشته باشد حمله خود را انجام می دهد. حمله های سیلابی[٤] نمونه این نوع حمله ها هستند که در آنها منابع محدودی مانند حافظه، ظرفیت پردازش پردازنده و پهنای باند خطوط انتقال به طور کامل در اختیار درخواستهای حمله کننده قرارگرفته و سرویس دهنده درخواستهای

بعدی کاربران مجاز به منظور استفاده از این منابع را رد می کند [۱۰].

در گونه دیگری از حمله های از کار اندازی سرویس حمله کننده با استفاده نا مناسب از سرویس های ارائه شده و اعمال پارامترهای خاص به توابعی که آن سرویس ها را اجرا می کنند باعث قطع شدن کلیه فعالیتهای سرویس دهنده می شوند [۱۱].

دسته دیگری از حمله های از کار اندازی سرویس به تغییر یا تخریب اطلاعات ساختاری سیستم می پردازند. این نوع از حمله ها مشابه حمله هایی است که محرمانگی و صحت اطلاعات را تهدید می کنند زیرا در هر دو اطلاعاتی از سیستم برداشت شده یا به سیستم تزریق می شود. در محرمانگی و صحت اطلاعات یک پروتکل خاص مسئول برقراری امنیت است و در آنها حمله کننده با به دست آوردن کلیدها و رمزهای موجود در پروتکل امنیت، آن را شکسته و سپس اطلاعاتی را به صورت غیر مجاز از سیستم برداشت کرده یا تزریق می کند. در تخریب یا تغییر اطلاعات ساختاری، حمله کننده با استفاده از ترتیب خاصی از سرویس های ارائه شده توسط سیستم، بدون آنکه به دنبال به دست آوردن کلید و رمز باشد، برخی پایگاه های داده سیستم را تغییر داده یا تخریب می کند [۱۲].

حمله های DoS از ضعف الگوریتم ها و پروتکل های موجود در شبکه و همچنین نقاط ضعف نرم افزارهای به کار گرفته شده برای پیاده سازی پروتکل ها در شبکه بهره برده و باعث از کار اندازی سیستم می شوند. تحلیل نمونه های مختلف این نوع حمله ها و دسته بندی آنها ما را قادر خواهد ساخت تا راه حلهای مشترکی برای مقابله با حمله های هر دسته استخراج کرده و اصولی برای طراحی پروتکل های مقاوم در برابر این حمله ها به دست آوریم.

در این مقاله یک دسته بندی از این حمله ها را ارائه کرده ایم. همچنین برخی اصول لازم برای طراحی پروتکل هایی که در مقابل این نوع حمله ها مقاوم باشند را مطرح کرده و آنها را تحلیل کرده ایم. در این مقاله همچنین نشان داده ایم که برخی از پروتکل های TCP/IP با اصول مطرح شده سازگار نیستند و به این دلیل در مقابل حمله های از کار اندازی سرویس آسیب پذیرند. همچنین در این مقاله برخی از حمله های از کار اندازی را بررسی کرده و

راه حل هایی برای سازگار کردن آنها با اصول مطرح شده ارائه کرده ایم.

ضرورت تحلیل و دسته بندی حمله های از کار اندازی سرویس

تقریبا تمام تحقیقات انجام شده بر روی موضوع امنیت در شبکه ها در دو زمینه محرمانگی و صحت اطلاعات انجام شده اند. در این مطالعات بیشتر از روشها و ابزار شکلی[۵] برای معین سازی[۶] و ارزیابی[۷] پروتکل های تصدیق هویت [۸] استفاده شده است [۱۳] و کارایی این روشها در پیدا کردن نقاط ضعف و ارائه پیشنهادهای سودمند برای رفع آنها به اثبات رسیده است. در مورد پروتکل های مربوط به محرمانگی و صحت اطلاعات M. Abadi و R. Needham با ارائه ۱۱ اصل نشان دادند که این اصول، شرایط لازم را برای درستی پروتکل های مذکور فراهم می آورند و اگر در پروتکلی هر یک از این اصول برقرار نباشد، آن پروتکل به درستی عمل نخواهد کرد [۱٤]. متعاقباً G. Lowe نشان داد [۱۵] که پروتکل Needham-Schroeder به دلیل آنکه اصل سوم از اصول یازده گانه مطرح شده فوق را دارا نیست قابل شکسته شدن است. او همچنین با استفاده از ابزار شکلی FDR اثبات کرد که با تغییر پروتکل مذکور به گونه ای که اصل مورد اشاره برقرار شود، امکان شکستن پروتکل اصلاح شده وجود نخواهد داشت [۱٦]. با دنبال کردن الگویی که برای موارد محرمانگی و صحت اطلاعات ایجاد شده است، در مورد قابلیت دسترسی نیز، به عنوان یکی از سه محور امنیت در شبکه ها، می خواهیم اصول لازم برای حصول قابلیت دسترسی را استخراج کنیم.

روش ما برای دستیابی به این اصول به این صورت است که ابتدا سعی خواهیم کرد با یک دسته بندی مناسب انواع حمله های از کار اندازی سرویس را در چندین دسته مشخص قرار دهیم، سپس با تحلیل حمله های موجود در هر دسته نقاط ضعف مشترک موجود در پروتکل ها را که منجر به موفقیت آمیز بودن حمله های یک دسته خاص می شوند استخراج کنیم و در نهایت اصولی ارائه دهیم که پروتکل ها در صورت عدم برقراری آن اصول در مقابل حمله های آن دسته خاص آسیب پذیر باشند. موضوع مهم آن است که این اصول کافی نبوده و

سازگار بودن پروتکل ها با آنها به معنای برقراری کامل قابلیت دسترسی نیست، بلکه سازگاری با آنها برای برقراری قابلیت دسترسی ضروری است. همچنین با کامل شدن مجموعه اصول لازم، استخراج نقاط ضعف پروتکل ها و نقاط ضعف موجود در راه حلهای ارائه شده برای مقابله با این حمله ها به سادگی انجام می شود.

یک دسته بندی از حمله های DoS تقسیم آنها به دو دسته زیر است:

۱ – آنهایی که از نقاط ضعف پروتکل ها استفاده می کنند، و

۲ – آنهایی که از نقاط ضعف نرم افزارها استفاده می کنند [۱۷].

در این دسته بندی برخی پروتکل های پرکاربرد مانند TCP و IP در نظر گرفته شده اند و برخی از انواع حمله ها را بر اساس این پروتکل ها دسته بندی کرده اند. این دسته بندی با در نظر گرفتن تمام پروتکل های TCP/IP حمله های مختلف را پوشش می دهد ولی ما را در یافتن اصول یاری نخواهد کرد زیرا تعداد این پروتکل ها بسیار زیاد بوده و حمله های متفاوتی بر علیه هر یک از آنها وجود دارد و حتی ممکن است دو حمله با دو مکانیزم مختلف بر علیه یک پروتکل خاص وجود داشته باشد.

در یک دسته بندی دیگر، حمله های از کار اندازی سرویس به دسته های

۱ – آنهایی که از منابع محدود و تجدید ناپذیر سیستم استفاده می کنند، و

۲ – آنهایی که به تخریب منابع می پردازند تقسیم شده اند [۱۸].

این دسته بندی برای دستیابی به اصول مناسب به نظر می رسد، با این حال انجام اندکی اصلاحات در آن ضروری است. در این دسته بندی، دسته دوم حمله هایی را در نظر می گیرد که اطلاعات موجود در یک سرویس دهنده را تخریب می کنند و حتی در این دسته حمله های فیزیکی مانند زمین کردن سیگنال نیز گنجانده شده است. نوعی از حمله ها وجود دارند که از محدود بودن منابع استفاده نکرده و صرفا با استفاده نامناسب از سرویس ها حمله را ترتیب می دهند. در دسته بندی بالا این نوع حمله ها در

نظر گرفته نشده اند.

با توسیع دسته بندی بالا، به گونه ای که تمام انواع حمله های از کار اندازی

سرویس پوشش داده شوند، دسته بندی جدید زیر را ارائه می کنیم:

۱ – آنهایی که منابع محدود را کاملا به خود اختصاص می دهند (اختصاص منبع)،

۲ – آنهایی که با استفاده نامناسب از سرویس ها به تخریب منابع می پردازند (تخریب منبع)، و

۳ – آنهایی که با استفاده از سرویس های ارائه شده توسط سیستم به تغییر یا تخریب اطلاعات ساختاری سیستم می پردازند (تغییر یا تخریب اطلاعات ساختاری).

با در اختیار داشتن چنین دسته بندی از حمله های DoS سعی خواهیم کرد اصول لازم برای طراحی پروتکل های مقاوم در مقابل حمله های هر یک از سه دسته بالا را ارائه دهیم.

اصول لازم پیشنهادی

یک نمونه از کارهای قبلی به منظور یافتن اصول لازم برای طراحی پروتکل های مقاوم در برابر حمله های DoS توسط Leiwo، Aura و Nikander انجام شده است [۱۹]. در آن مقاله اصولی ارائه شده اند که در آنها مفاهیم تصدیق هویت و تشخیص حمله به عنوان راه حل مقابله با حمله های از کار اندازی سرویس پیشنهاد شده اند. به کارگیری این روشها خود تهدیدهای جدیدی در مقابل حمله های DoS ایجاد خواهد کرد. به عنوان مثال یکی از اصول ارائه شده آن است که "قبل از اختصاص هرگونه منبع به یک درخواست باید درخواست کننده تصدیق هویت شود". برقراری این اصل به معنای استفاده از یک منبع محدود دیگر در سیستم به منظور کنترل دسترسی است که خود ممکن است دچار حمله های از کار اندازی سرویس شود [۲۰]. برخی اصول ارائه شده دیگر تغییرات بنیادی در سیستم را ضروری می سازند. به عنوان نمونه "سیستم باید به گونه ای باشد که هرگونه استفاده از یک منبع خاص سرویس دهنده استفاده بیشتری از یکی از منابع متعلق به درخواست کننده آن منبع را منجر شود". علاوه بر مشکلاتی که در پیاده سازی اصول ارائه شده در آن مقاله وجود دارد، اصول ارائه شده فقط حمله های اختصاص منابع را در نظر گرفته و دیگر انواع

حمله های DoS را مورد بحث قرار نداده اند.

در این قسمت اصول پیشنهادی خود را مطرح کرده و کارایی آنها را در مقاوم کردن پروتکل ها در مقابل حمله های DoS تحلیل خواهیم کرد.

حمله های اختصاص منبع

در این نوع حمله ها، حمله کننده با اختصاص تکراری و جداگانه یک منبع آن را کاملا برای در خواستهای خود اختصاص می دهد و به این ترتیب درخواستهای کاربران مجاز برای در اختیار گرفتن قسمتی از این منبع، توسط سرویس دهنده رد خواهد شد. به عبارت دیگر سرویسی که از آن منبع استفاده می کند غیر قابل دسترس خواهد بود. اولین اصل پیشنهادی خود را با نام اصل **"منبع قابل دسترس"** به شکل زیر مطرح می کنیم.

فرض کنیم منبع قابل دسترس در زمان t با $R_{available}(t)$ نشان داده شود، در این صورت اصل منبع قابل دسترس بیان می کند که در هر پروتکل مقاوم در برابر حمله های اختصاص منبع در هر لحظه t باید دو شرط زیر برقرار باشد:

$$R_{available}(t) > \alpha \text{ و } |\frac{\partial R_{available}(t)}{\partial t}| < \beta \quad (۱)$$

به عبارت دیگر اولا باید در هر لحظه مقدار منبع قابل دسترس از مقدار مشخصی بزرگتر باشد و ثانیا میزان تغییرات مقدار منبع قابل دسترس در هر لحظه از مقدار مشخصی کمتر باشد. مقادیر α و β بر حسب مشخصات سیستم و همچنین مدل در نظر گرفته شده برای حمله کننده محاسبه می شوند.

حمله کننده به ازای هر درخواستی که به سرویس دهنده می فرستد به میزان Δr از منبع را به خود اختصاص می دهد. در صورتی که زمان کافی برای سرویس دهنده وجود نداشته باشد تا منبع اختصاص داده شده به درخواستهای حمله کننده را آزاد کند، با ارسال n درخواست به طوری که $n > \frac{R_0}{\Delta r}$ آن منبع برای درخواستهای بعدی قابل دسترس نخواهد بود (R_0 کل مقدار منبع است و فرض کرده ایم $R_{available}(0) = R_0$). بنابر این در تحلیل این اصل، مقدار زمانی که مقدار منبع Δr در اختصاص حمله کننده

باقی می ماند به عنوان یک متغیر تصادفی در نظر گرفته شده و در تعیین پارامترها نقش مهمی دارد. اگر متغیر تصادفی زمان در اختیار بودن Δr را با $\Delta\tau$ نشان دهیم یکی از روشهای مقابله با این نوع حمله ها کاهش متوسط $\Delta\tau$ است (در تمام روابطی که در این مقاله آمده است از حروف پررنگ برای بیان متغیر تصادفی و از شکل معمولی همان حروف برای بیان مقدار متغیر تصادفی استفاده شده است).

حمله کننده با یک توزیع تصادفی خاص درخواستهایی به سرویس دهنده ارسال می کند، همچنین تعداد کل درخواستهای کاربران معمولی از توزیع تصادفی پوآسن با پارامتر λ درخواست در واحد زمان پیروی می کند. هر درخواست کاربر معمولی در فاصله زمانی ΔT مقدار منبع Δr را به خود اختصاص می دهد. مقدار ΔT نیز یک متغیر تصادفی است و فرض خواهیم کرد که با تقریب از توزیع گاوسی پیروی می کند. بنابراین مقدار منبع قابل دسترس در زمان t از رابطه زیر به دست می آید.

$$R_{available}(t) = R_0 - y_1(t)\Delta r - y_2(t)\Delta r \qquad (۲)$$

در این رابطه $y_1(t)$ تعداد درخواستهایی از کاربران معمولی جهت در اختیار گرفتن منبع است که در فاصله زمانی $[0,t)$ اعلام شده است ولی در همین فاصله زمانی مقدار منبع در اختیار گرفته شده را آزاد نکرده اند و $y_2(t)$ تعداد درخواستهایی از حمله کننده در این فاصله است که در این فاصله منبع در اختیار خود را آزاد نکرده اند. برای تحلیل دقیقتر، حالتهای زیر را در نظر می گیریم:

حالت ۱- حالتی را در نظر بگیریم که مقدار متغیر تصادفی $\Delta\tau$، $\Delta\tau$، مقدار ثابت بزرگی باشد. در این صورت حداقل یک $t_0 \in \Re$ ($\Re$ مجموعه اعداد حقیقی است) وجود خواهد داشت که $R_{available}(t_0) = 0$. این موضوع نشان می دهد در صورتی که مکانیزمی برای محدود کردن $\Delta\tau$ اتخاذ نگردد حصول قابلیت دسترسی امکان پذیر نخواهد بود.

حالت ۲- حالتی را در نظر بگیریم که $\Delta\tau$ دارای توزیع چگالی احتمال $f_{\Delta\tau}(\Delta\tau)$ باشد. در این حالت باید توزیع تصادفی $\mathbf{y_1(t)}$ و $\mathbf{y_2(t)}$ را استخراج کنیم. ابتدا به محاسبه توزیع $\mathbf{y_1(t)}$ می پردازیم. تعداد n درخواست تحت توزیع پوآسن تولید شده اند که پارامتر این توزیع λt است. این n درخواست

با یک توزیع یکنواخت در فاصله زمانی $[0,t)$ توزیع می شوند و متغیر این توزیع را $\boldsymbol{\theta}$ فرض می کنیم. برای یافتن توزیع $\mathbf{y_1(t)}$ باید احتمال آن را حساب کنیم که تعداد y_1 درخواست از این n درخواست قبل از زمان t منبع در اختیار خود را آزاد نکنند. بنابراین احتمال اینکه یک درخواست در فاصله مذکور رخ دهد و فضای در اختیار خود را آزاد کند برابر است با:

$$A(t)=\int_{x=0}^{t}P(\boldsymbol{\theta}=x)G(\frac{t-x-\mu}{\sigma})dx=\frac{1}{t}\int_{x=-\mu/\sigma}^{(t-\mu)/\sigma}G(x)dx$$

(۳)

که در آن تابع $G(x)$ تابع توزیع گاوسی با متوسط صفر و انحراف معیار واحد است و همچنین فرض کرده ایم که متوسط و انحراف معیار متغیر تصادفی $\mathbf{\Delta T}$ به ترتیب μ و σ است. بنابراین تابع چگالی احتمال $\mathbf{y_1(t)}$ از رابطه زیر بـــــــه دست می آید.

$$f_{\mathbf{y_1(t)}}(y_1(t))=\sum_{n=o}^{\infty}\sum_{i=0}^{n}C(n,y_1(t))(A(t))^{n-y_1(t)}(1-A(t))^{y_1(t)}$$

$$.\frac{\exp(-\lambda t)(\lambda t)^n}{n!}\delta(y_1(t)-i)$$

(٤)

که در آن $\delta(x)$ تابع ضربه[۹] و $C(n,m)$ ترکیب m از n است. فرض کنیم تعداد درخواستهای حمله کننده تابعی معین از طول فاصله زمانی مانند $m(t)$ باشد و این تعداد با توزیع یکنواخت در فاصله مذکور توزیع شده باشند. در این حالت تابع چگالی احتمال $\mathbf{y_2(t)}$ به شکل زیر خواهد بود.

$$f_{\mathbf{y_2(t)}}(y_2(t))=\sum_{i=0}^{m(t)}C(m(t),y_2(t))(B(t))^{m(t)-y_2(t)}$$

$$.(1-B(t))^{y_2(t)}\delta(y_2(t)-i)$$

(۵)

که در آن :

$$B(t)=\frac{1}{t}\int_{x=0}^{t}F_{\mathbf{\Delta\tau}}(t-x)dx\ ,\ F_{\mathbf{\Delta\tau}}(x)=\int_{y=0}^{x}f_{\mathbf{\Delta\tau}}(y)dy$$

(٦)

اگر مقدار منبع قابل دسترس در زمان t را یک متغیـــــر تصادفی به شکل $\mathbf{R(t)}=R_0-\mathbf{y_1(t)}\Delta r-\mathbf{y_2(t)}\Delta r$ در نظر بگیریم و برای سادگی روابط از نوشتن آرگومان t صرف نظر کنیم، تابع چگالی احتمال $\mathbf{R(t)}$ به شکل زیر خواهد بود.

$$f_{\mathbf{R}}(R)=\frac{1}{\Delta r}\int_{y_2=0}^{+\infty}f_{\mathbf{y_1}}((R_0-R-y_2\Delta r)/\Delta r)f_{\mathbf{y_2}}(y_2)dy_2$$

(۷)

لازم به ذکر است که رابطه (۷) با فرض مستقل بودن متغیرهای تصادفی $\mathbf{y_1(t)}$ و $\mathbf{y_2(t)}$ به دست آمده است و در صورتی که راه

حلهای ارائه شده برای مقابله با حمله های اختصاص منابع، زمان در اختیار داشتن منابع توسط کاربران معمولی را نیز تحت تاثیر قرار دهند باید از تابع چگالی مشترک این دو متغیر استفاده کرد.

با انتخاب مناسب تابع چگالی احتمال $f_{\Delta\tau}(\Delta\tau)$ می توان اصل منبع قابل دسترس را برقرار ساخت. به عبارت دیگر باید این تابع چنان انتخاب شود که شرایط زیر برقرار باشد.

$$P(\mathbf{R(t)} > \alpha) = P(|\, d\mathbf{R(t)}/dt \,| < \beta) = 1 \qquad (۸)$$

در صورتی که اصل منبع قابل دسترس با شرایط در اختیار گرفته شدن برخی منابع به کار گرفته شده در یک پروتکل سازگار نباشد آن پروتکل در مقابل حمله های اختصاص منبع آسیب پذیر است و برای مقاوم کردن آن پروتکل و همچنین سازگار کردن آن با اصل مذکور راه حلی ارائه خواهد شد. این راه حل تابع چگالی $f_{\Delta\tau}(\Delta\tau)$ را نسبت به آنچه در قبل از ارائه راه حل وجود داشته است تغییر می دهد ولی ممکن است شکل تابع چگالی زمان در اختیار بودن منابع توسط کاربران معمولی را نیز تغییر دهد. این موضوع ممکن است تهدید جدیدی در قابلیت دسترسی منابع برای کاربران معمولی ایجاد کند. اگر فرض کنیم تابع چگالی احتمال زمان در اختیار بودن منابع توسط کاربران معمولی پس از ارائه راه حل $f^{*}_{\Delta\mathbf{T}}(\Delta T)$ و قبل از آن $f_{\Delta\mathbf{T}}(\Delta T)$ باشد، کمیت زیر نشان دهنده موفقیت راه حل ارائه شده در مقابله با حمله های اختصاص منابع است.

$$E = \int_{\Delta T=-\infty}^{+\infty} (f^{*}_{\Delta\mathbf{T}}(\Delta T) - f_{\Delta\mathbf{T}}(\Delta T))^2 \, d(\Delta T) \qquad (۹)$$

در حالی که اصل منبع قابل دسترس برقرار است، هرچه مقدار E کوچکتر باشد به معنای آن است که راه حل ارائه شده قابلیت دسترسی کاربران معمولی را افزایش و قابلیت دسترسی حمله کننده را کاهش داده است.

به عنوان مثال حمله SYN Flooding را در نظر می گیریم. این حمله از نقاط ضعف پروتکل TCP ناشی می شود. در این پروتکل قبل از اینکه دیتا از مبدا به مقصد فرستاده شود

نیاز به ایجاد یک اتصال از مبدا به مقصد است. فرآیند ایجاد چنین اتصالی شامل سه مرحله است: در مرحله اول فرستنده یک بسته SYN به مقصد می فرستد، سپس مقصد پس از دریافت SYN بسته ای شامل SYN و ACK برای مبدا می فرستد و در پایان مبدا یک ACK برای مقصد می فرستد. مقدار مشخصی حافظه برای هر اتصال TCP در نظر گرفته می شود که البته این مقدار در گونه های مختلف سیستم عامل متفاوت است [۲۱] . حالت اتصال نیمه باز حالتی است که SYN از مبدا به مقصد ارسال شده است و مقصد نیز پیغام SYN+ACK را صادر کرده است ولی هنوز ACK از مبدا به مقصد ارسال نشده است. در این حالت حافظه ذکر شده اختصاص یافته است و مقصد مقدار زمان مشخصی منتظر می ماند. در صورتی که پس از سپری شدن این زمان ACK از مبدا به مقصد فرستاده نشود یا اینکه سرویس دهنده آن را دریافت نکند، اتصال قطع شده و حافظه آزاد می شود. به این ترتیب اگر تعداد اتصالات نیمه باز از حد مشخصی بیشتر شود سرویس دهنده به دلیل اشغال شدن تمام فضای حافظه درخواست های بعدی را رد می کند و به این ترتیب قابلیت دسترسی سرویس برقراری اتصال یا اختصاص حافظه از بین می رود.

اگر حداکثر زمان انتظار سرویس دهنده به منظور دریافت ACK از منبع را T_0 فرض کنیم. تابع زمان در اختیار بودن حافظه برای کاربران معمولی را می توان به صورت توزیع گاوسی بریده شده در T_0 در نظر گرفت. به این ترتیب تابع چگالی زمان در اختیار بودن حافظه برای کاربران معمولی به صورت زیر است.

$$f_{\mathbf{\Delta T}}(\Delta T) = \frac{1}{c\sqrt{2\pi\sigma}} \exp((\Delta T - \mu)^2 /(2\sigma^2)) \ , \ 0 \le \Delta T \le T_0$$

$$c = G((T_0 - \mu)/\sigma)) - G(-\mu/\sigma)$$

(۱۰)

همچنین در این حالت تابع چگالی زمان در اختیار بودن حافظه برای حمله کننده به شکل $f_{\mathbf{\Delta\tau}}(\Delta\tau) = \delta(\Delta\tau - T_0)$ است. به علاوه فرض کنیم حمله کننده در واحد زمان q درخواست برای برقراری اتصال صادر کند. در این حالت:

$$B(t) = \begin{cases} (t - T_0)/t & t \ge T_0 \\ 0 & t < T_0 \end{cases}$$

(۱۱)

بنابراین تابع چگالی احتمال تعداد درخواستهای حمله کننده که در فاصله $[0,t)$ ایجاد و در این فاصله حافظه در اختیار خود را آزاد نمی کنند برای $t > T_0$ به شکل زیر است.

$$f_{y_2}(y_2) = \sum_{i=0}^{qt} C(qt, y_2)(\frac{t-T_0}{t})^{qt-y_2}(\frac{T_0}{t})^{y_2}\delta(y_2 - i) \quad (۱۲)$$

بر اساس رابطه (۱۲) هرچه t بزرگتر می شود احتمال رخ دادن مقادیر بزرگتر y_2 بیشتر می شود و به این ترتیب با افزایش زمان، مقدار حافظه در اختیار حمله کننده بیشتر خواهد بود و هرچه مقدار q بزرگتر باشد این مقدار سریعتر به یک نزدیک می شود.

با توجه به روابط فوق اصل منبع قابل دسترس در پروتکل TCP برقرار نیست. یک راه حل برای برقراری این اصل در سرویس برقراری اتصال TCP آن است که در صورت کمتر شدن حافظه قابل دسترس از یک مقدار معین یا بیشتر بودن میزان تغییرات آن از یک مقدار خاص، به ازای دریافت هر درخواست برقراری اتصال آن درخواستی را که طولانی ترین زمان انتظار را داشته است بیرون بریزد و فضای حافظه مربوط به آن را آزاد کند. البته با این تغییر برخی درخواستهای کاربران معمولی که فاصله زمانی ارسال ACK آنها طولانی است رد خواهد شد. به عبارت دیگر تابع چگالی زمان در اختیار داشتن حافظه برای آنان تغییر می کند و مقدار E صفر نخواهد بود. راه حلی که ضمن برقراری اصل منبع قابل دسترس کوچکترین مقدار E را داشته باشد به عنوان بهترین راه حل شناخته می شود.

در شکل (۱) قسمتی از نمودار حالت برقراری اتصال TCP را که به منظور برقرار بودن اصل منبع قابل دسترس اصلاح شده است رسم کرده ایم. در صورتی که شرایط لازم برای برقراری اصل منبع قابل دسترس نقض شود سرویس دهنده یک اخطار حافظه با نام MA [۱۰] ایجاد می کند. همچنین سرویس دهنده در هر لحظه پیغام MWC [۱۱] را برای ماشین حالت متناظر با اتصالی که بیشترین زمان انتظار قبل از دریافت ACK را داشته است ارسال می کند. به این ترتیب اگر در ماشین حالت متناظر با یک اتصال MA+MWC دریافت شود و ماشین در حالت SYN

recvd باشد، از آن حالت به حالت Listen انتقال پیدا کرده و حافظه متناظر بـــا اتصال آزاد می شود و اتصال مجددا راه اندازی[۱۲] می شود.

فرض کنیم x متغیر تصادفی تعداد درخواستها برای برقراری اتصال TCP در یک فاصله زمانی معین باشد. این متغیر تصادفی مجموع دو متغیر تصادفی x_1 و x_2 است. متغیر تصادفی x_1 بیان کننده تعداد درخواستهایی است که در آنها طول فاصله زمانی مابین رسیدن SYN و ACK، متناظر با آن درخواست، به سرویس دهنده از حداکثر زمان انتظار سرویس دهنده، T_0، کمتر است. متغیر تصادفی x_2 بیان کننده تعداد درخواستهایی است که طول این فاصله زمانی برای آنها برابر T_0 است (علاوه بر درخواستهای حمله کننده ممکن است برخی درخواستهای کاربران معمولی نیز در این دسته قرار گیرد). فرض کنیم x_1 و x_2 به ترتیب از توزیع پوآسن با پارامترهای λ_1 و λ_2 پیروی کنند. در این صورت اگر متوسط زمان در اختیار بودن منبع را قبل از اعمال راه حل بالا با $\overline{t_b}$ و پس از اعمال راه حل بالا با $\overline{t_a}$ نشان دهیم، روابط زیر برقرار است.

$$\overline{t_b} \cong \frac{\lambda_1}{\lambda_1+\lambda_2} E(\mathbf{t_1}) + \frac{\lambda_2}{\lambda_1+\lambda_2} T_0 \qquad (۱۳)$$

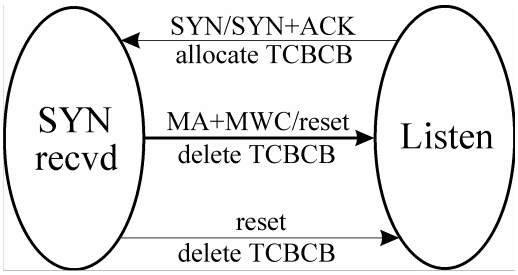
$$\overline{t_a} \cong \frac{\lambda_1}{\lambda_1+\lambda_2} E(\mathbf{t_1'}) + \frac{\lambda_2}{\lambda_1+\lambda_2} (T_0 - T_I) \qquad (۱٤)$$

در این روابط $E(\mathbf{t_1})$ و $E(\mathbf{t_1'})$ به ترتیب بیان کننده امید ریاضی متغیر تصادفی زمان در اختیار بودن منبع به ازای درخواستهای x_1 قبل و بعد از اعمال راه حل بالا است. همچنین T_I بیان کننده میزان کاهش متوسط زمان انتظار سرویس دهنده به ازای درخواستهایی است که با متغیر x_2 مشخص می شوند. بنابراین با این فرض که $E(\mathbf{t_1}) = E(\mathbf{t_1'})$:

$$\overline{t_b} - \overline{t_a} \cong \frac{\lambda_2}{\lambda_1+\lambda_2} T_I . \qquad (۱۵)$$

فرض $E(\mathbf{t_1}) = E(\mathbf{t_1'})$ هنگامی برقرار است که حمله کننده برای ارسال درخواستهای خود محدودیت هایی داشته باشد. این محدودیت ممکن است به دلیل محدود بودن سرعت خطوط ارتباطی باشد. بنابر این در صورتی که چنین محدودیتی وجود نداشته باشد راه حل ارائه شده فوق منجر به اصلاح قابلیت

دسترسی نخواهد شد. این موضوع به دلیل بزرگ شدن مقدار E در رابطه (۹) و در نتیجه تغییر مقدار $E(\mathbf{t}_1')$ در رابطه (۱٤) است.



شکل ۱ : نمودار حالت اصلاح شده TCP در حالتهای SYN recvd و Listen .

حمله های تخریب منبع

در حمله های تخریب منبع حمله کننده با استفاده نامناسب از سرویس های ارائه شده توسط شبکه و اعمال پارامترهای خاص به توابعی که وظیفه اجرای یک سرویس خاص را دارند باعث می شود فعالیتهای سرویس دهنده متوقف شود. به عبارت دیگر حمله کننده سرویس دهنده را وادار می کند تا یک سرویس خاص را در شرایط ویژه ای اجرا کند که منجر به تخریب برخی منابع خواهد شد. بنابراین در صورتی که برای هر یک از سرویس های سیستم، که از یک یا چند منبع جهت اجرای سرویس استفاده می کنند، شرایط اجرای صحیح را مشخص کنیم، در حمله های تخریب منبع برخی از این شرایط نقض می شوند.

برای طراحی پروتکل هایی که در مقابل این نوع حمله ها مقاوم باشند **"اصل اجرای مستقل"** را به شکل زیر مطرح می کنیم. اصل اجرای مستقل بیان می کند که پروتکل مقاوم در مقابل حمله های تخریب منبع دارای سرویس هایی است که هریک از آنها مستقل از بسته های اطلاعات و کنترلی که از طریق حد فاصل وارد سیستم می شوند همواره تحت شرایط یکسانی، به نـام شرایط اجرای صحیح،
اجرا می شوند.

اگر سرویس به گونه ای اجرا شود که برخی شرایط موجود در شرایط اجرای صحیح آن سرویس نقض شوند برخی منابع سیستم تخریب خواهند شد. سازگاری یک پروتکل با اصل اجرای مستقل به آن معناست که حتی در صورت اعمال پارامترهایی که منجر به اجرای مخرب سرویس خواهند شد

سیستم قادر است سرویس را به صورتی اجرا کند که منجر به تخریب منابع نشود.
فرض کنیم حمله کننده بسیار قوی است، یعنی قادر است بسته های اطلاعاتی و کنترلی با هر شکل و هر تعداد را به هدف مورد حمله خود ارسال کند (به دلیل عدم وجود یک مدل دقیق از تواناییهای حمله کننده در شبکه های کامپیوتری این فرض به عنوان یک فرض محافظه کارانه برای طراحی پروتکل های مقاوم در مقابل حمله های تخریب منبع ضروری است). در این حالت، در صورت وجود شرایطی برای اجرای صحیح سرویس، حمله کننده قادر است آن شرایط را نقض کرده و حمله خود را انجام دهد. بنابراین در صورتی که یک سرویس با این اصل سازگار نباشد می توان یک الگوریتم پردازشی به آن سرویس اضافه کرد به گونه ای که قسمت اضافه شده دارای مجموعه شرایط اجرای صحیح تهی باشد و چنان عمل کند که مجموعه شرایط اجرای صحیح کل آن سرویس (با در نظر گرفتن قسمت اضافه شده) برای ناظر بیرونی سیستم، از جمله حمله کننده، یک مجموعه تهی باشد. بنابراین اصل اجرای مستقل معادل با تهی بودن مجموعه شرایط اجرای صحیح است و در این شرایط ارسال هرگونه بسته اطلاعاتی منجر به تخریب منبع نخواهد شد.

به عنوان مثال سرویس Ping را در نظر می گیریم. این سرویس یکی از سرویس های موجود در پروتکل ICMP است. مجموعه شرایط اجرای صحیح برای این سرویس شامل دو شرط زیر است: ۱- طول بسته Ping باید کوچکتر از ٦٥٥٣٥ باشد، و ۲- در صورتی که برای اجرای این سرویس از فرآیند خرد سازی بسته های اطلاعاتی در مبدا و یکپارچه سازی آنها در مقصد استفاده می شود، طول تکه ها باید مثبت باشد (حوزه آفست بسته عددی مثبت باشد). ارسال بسته های Ping توسط حمله کننده به گونه ای که حداقل یکی از این شرایط نقض شوند باعث از کار افتادن سیستم مقصد می شود. در نتیجه سرویس Ping یک سرویس آسیب پذیر است. اگر فرآیندی در سیستمی که بسته های Ping را دریافت می کند، ۱- بسته های خرد شده Ping را بیرون بریزد (با توجه به اینکه در ارسال بسته های بزرگتر از ٦٥٥٣٥ بایت لزوما از فرآیندهای خرد سازی و یکپارچه سازی استفاده می شود)، و ۲- بسته های خرد

نشده ای را که دارای آفست منفی هستند بیرون بریزد، شرایط اجرای صحیح این سرویس تهی خواهد شد. به عبارت دیگر مستقل از شکل و محتوای بسته های ارسال شده توسط حمله کننده، این سرویس همواره به گونه ای اجرا می شود که منجر به تخریب منابع نخواهد شد. همچنین باید سرویس افزوده شده خود دارای مجموعه تهی از شرایط اجرای صحیح باشد.

راه حل بالا مقاومت سیستم در مقابل این حمله را تضمین می کند ولی قابلیت دسترسی این سرویس را برای کاربران معمولی محدود می کند، زیرا در این صورت ارسال بسته های Ping خرد شده توسط کاربران معمولی که حتی شرایط اجرای صحیح را نیز رعایت کرده باشند بدون پاسخ خواهد ماند (این نوع سرویس به عنوان یک سرویس مجاز در پروتکل ICMP تعریف شده است). برای رفع این مشکل می توان فرآیند اضافه شده به سرویس را به گونه ای طراحی کرد که با دریافت اطلاعات مربوط به وضعیت[۱۲] حافظه ای که وظیفه ذخیره سازی بسته های دریافتی را دارد تشخیص دهد که آیا مجاز به دریافت تکه خرد شده بعدی و ذخیره سازی آن در حافظه است. در این صورت قابلیت دسترسی سرویس برای کاربران معمولی تضمین می شود.

برای طراحی پروتکل های مقاوم در مقابل حمله های تخریب منبع باید تمامی عناصر مجموعه شرایط اجرای صحیح کاملا معین شوند. این موضوع ذاتا مساله دشواری است و این به دلیل آن است که با وقوع حمله های جدید ممکن است عناصر جدیدی به مجموعه شرایط اجرای صحیح اضافه شوند. پیدا کردن روش سیستماتیک برای یافتن عناصر این مجموعه موضوعی در خور تحقیق است.

روش عمومی برای مقاوم کردن پروتکل ها در مقابل حمله های تخریب منبع استفاده از فرآیند پیشگیری در جلوی سرویس آسیب پذیر[۱۳] است. این فرآیند را به طور اختصاری با PFEP نشان می دهیم. فرآیند PFEP باید دارای دو ویژگی زیر باشد.

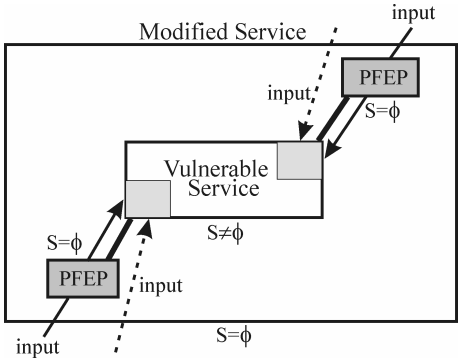
۱ – بسته های رسیده را با توجه به شرایط موجود در مجموعه شرایط اجرای صحیح سرویس چک کند و سپس بر اساس اطلاعات وضعیت دریافتی از سرویس

آسیب پـــــذیر، بسته های بد شکل را بیرون بریزد.

۲ – مجموعه شرایط اجرای صحیح PFEP باید تهی باشد. این بدان معنی است که فرآیند پیشگیری خود باید کاملا در مقابل حمله های تخریب منبع مقاوم باشد.

این موضوع بیان کننده آن است که هیچ تغییر عمده ای به جز اضافه کردن PFEP و اندکی تغییرات مختصر در سرویس آسیب پذیر به منظور انتقال اطلاعات وضعیت به فرآیند پیشگیری لازم نیست. به علاوه ساختن یک PFEP با مجموعه شرایط اجرای تهی ساده است زیرا وظیفه آن فقط ارزیابی هماهنگی بسته های دریافتی با عناصر مجموعه شرایط اجرای صحیح است و اتخاذ عمل مناسب متناسب با اطلاعات وضعیت رسیده از سرویس آسیب پذیر است.

شکل (۲) مفهوم ساختن یک سرویس مقاوم در مقابل حمله های تخریب منبع را با استفاده از سرویس آسیب پذیر و PFEP نشان می دهد.



شکل ۲ : مفهوم ساختن یک سرویس مقاوم در مقابل حمله های تخریب منبع با استفاده از سرویس آسیب پذیر و PFEP. مجموعه S نشان دهنده مجموعه شرایط اجرای صحیح است. پیکان نقطه چین، ورودی به سرویس اصلاح نشده و پیکان یکپارچه، ورودی به سرویس اصلاح شده را نشان می دهد. خط ضخیم بین PFEP و سرویس آسیب پذیر نشان دهنده عنصر انتقال دهنده اطلاعات وضعیت بین PFEP و سرویس آسیب پذیر است.

نتیجه گیری و جمع بندی

در این مقاله دسته بندی جدیدی از حمله های از کار اندازی سرویس ارائه شده است که در آن حمله های DoS به سه دسته "اختصاص منبع"، "تخریب منبع" و "تخریب یا تغییر اطلاعات ساختاری

سیستم" تقسیم شده اند. سپس دو اصل مهم برای طراحی پروتکل های مقاوم در برابر حمله های اختصاص منبع و تخریب منبع مطرح شده است. عدم برقراری این اصول، پروتکل را در مقابل چنین حمله هایی آسیب پذیر می سازد. همچنین اصل اختصاص منبع را به صورت آماری مورد تحلیل قرار داده ایم. به علاوه مشخص شد که در مجموعه پروتکل های TCP/IP برخی پروتکل ها مانند پروتکل های TCP و ICMP با این اصول سازگار نیستند و برای برقراری قابلیت دسترسی در این پروتکل ها لازم است برخی تغییرات ساختاری در آنها انجام شود. برخی از این تغییرات را در این مقاله پیشنهاد داده ایم.

مراجع

1 -Wulf, L. (1999). *Interaction and security in distributed computing.* [Online]. Available: http://web.comlab.ox.ac.uk/oucl/publications/books/concurrency/textmat/.

2 - McCullagh, D. (2000). *Was yahoo smurfed or trinooed.* [Online]. Available: http://www.wired.com/news/business/0,1367,34203,00.html.

3 -CERT/CC. (1997). *An analysis of security incidents on the Internet 1989-1995.* [Online]. Available: http://www.cert.org/research/JHThesis/start.html.

4 -CERT Advisory. (1997). *IP denial-of-service attacks.* CA-97.28.

5 -CERT Advisory. (2001). *File globing vulnerabilities in various FTP servers.* CA-2001.07.

6 -CERT Advisory. (1996). *BIND version 4.9.3.* CA-96.02.

7 -CERT Advisory. (2001). *Sadmind/IIS worm.* CA-2001-11.

8 -CERT Advisory. (2001). *Buffer overflow in Sun Solaris in.lpd print daemon.* CA-2001-15.

9 -CERT Advisory. (2001). *Oracle 8i contains buffer overflow in TNS listener.* CA-2001-16.

10 - CERT Advisory. (1996). *TCP SYN Flooding and IP spoofing attacks.* [Online]. Available: http://www.cert.org.

11 - CERT Advisory. (1996). *Denial-of service via ping*. CA-96.26.

12 - Felten, E. W., Balfanz, D., Dean, D. and Wallach, D. S. (1997). *Web spoofing: An Internet con game*. Department of Computer Science, Princeton University, Tech. Rep., 540-96. [Online]. Available: http://www.cs. princeton.edu/ sip/pub/spoofing.html.

13 - Wing, J. (1998). *A symbiotic relationship between formal methods and security*. School of Computer Science, Carnegie Mellon University, *Tech. Rep*. CMU-CS-98-188.

14 - Abadi, M. and Needham, R. (1996). "Prudent engineering practice for cryptographic protocols." *IEEE Trans. on Software Engineering*, Vol. 22, No. 1, PP. 6-15.

15 - Lowe, G. (1995). "An attack on the Needham-Schroeder public key authentication protocol." *Information Processing Letters*, Vol. 56, PP. 131-135.

16 - Lowe, G. (1996). "Breaking and fixing the Needham-Schroeder public key protocol using CSP and FDR." *Proceedings of 2nd TACAS Conference*, *Lecture Notes in Computer Science*, Vol. 36, PP. 147-166.

17 - Hautio, J. and Weckstorm, T. (1999). *Denial of service attacks*. Helsinki. [Online]. Available: http://www.hut.fi/u/tweckstr/hakkeri/ Dos_paper.html .

18 - CERT/CC. (2001). *Denial of service attacks*. [Online]. Available: http://www.cert.org/tech_tips/denial-of-service.html .

19 - Leiwo, J., Aura, T. and Nikander, P. (2000). "Towards network denial of service resistant protocols." *Proc. IFIP Sec.*, 2000. [Online]. Available: http://saturn.hut.fi/ publications/papers/aura/leiwo-nikander-aura-ifipsec00-abstract.html.

20 - Meadows, C. (1999). "A formal framework and evaluation method for network denial of service." *in Proc. of 12th IEEE Computer Security Foundations Workshop*, Morando, Italy, June1999, PP. 4-13, IEEE Computer Society Press.

21 - Postel, J. (1982). *Transmission Control Protocol*. [Online]. Available: http://www.freesoft.org/CIE/RFC/793/index.htm.

واژه های انگلیسی به ترتیب استفاده در متن

1 – Denial of Service

2 – Crash

3 – Reset

4 – Flooding

5 – Formal

6 – Specification

7 – Verification

8 – Authentication

9 - Impulse

10 – Memory Alarm

11 – Maximum Waited Connection

12 – Status

13 – Preventive-Front-End-Process