بررسی جنبه‌های امنیتی در مؤلفه‌ها و معماری جویش‌گرهای بومی

حسن کوشککی[1*]، شقایق نادری[2]، نسرین تاج نیشابوری[3] و مهسا امیدوار سرکندی[4]

[1]کارشناس ارشد امنیت مرکز تحقیقات مخابرات ایران، پژوهشکده امنیت، تهران، ایران
hassan.kooshkaki@gmail.com
[2]عضو هیأت علمی مرکز تحقیقات مخابرات ایران، پژوهشکده امنیت، تهران، ایران
naderi@itrc.ac.ir
[3]کارشناس ارشد امنیت مرکز تحقیقات مخابرات ایران، پژوهشکده امنیت، تهران، ایران
taj.ict@gmail.com
[4]کارشناس امنیت مرکز تحقیقات مخابرات ایران، پژوهشکده امنیت، تهران، ایران
mahsa.omidvarsarkandi@gmail.com

چکیده

امروزه با رشد شبکه‌های رایانه‌ای و افزایش فزاینده محتوا روی اینترنت، تقاضا برای تماشا و جستجوی محتواهایی مثل ویدیو، موسیقی، فایل‌ها و اسناد زیاد شده است و جویش‌گرها یکی از عواملی هستند که در زمینه جستجو به هر تقاضایی از طرف کاربران پاسخ می‌دهند و به آنها کمک می‌کنند تا سریع‌تر به اهداف خود برسند. جویش‌گرها علاوه‌بر مزیت‌های بسیاری که دارند درصورت طراحی و پیکربندی ضعیف مؤلفه‌های خزش‌گر، نمایه‌ساز و رتبه‌بند، عملکرد نادرستی خواهند داشت و منجر به افشای اطلاعات محرمانه کاربران و وب‌گاه‌ها، ارائه نتایج غیر مرتبط و آلوده و غیر امن‌شدن معماری جویش‌گر می‌شوند؛ از این‌رو توجه به مسائل و جنبه‌های امنیتی مؤلفه‌ها در معماری جویش‌گرها، از جمله جویش‌گرهای بومی یکی از الزامات اساسی برای آنها محسوب می‌شود. مهم‌ترین بخش‌هایی که باید امنیت آنها در معماری جویش‌گرهای بومی تضمین شود، شامل مؤلفه خزش‌گر (سیاست‌های خزش‌گر، فاکتورهای طراحی، حملات تزریق کد، دسترس‌پذیری خزش‌گرها و پایگاه داده خزش‌گرها)، مؤلفه نمایه‌ساز (واحد جستجو، پایگاه داده نمایه‌ساز، سازوکارهای نمایه‌سازی، فاکتورهای طراحی) و مؤلفه رتبه‌بند (سیاست‌های رتبه‌بندی، سیاست‌های دفاعی در برابر وب سایت‌های آلوده، بهینه‌سازی منفی جویش‌گرها یا سئو منفی یا به‌عبارت بهتر سئو کلاه‌سیاه) است. محورهای اصلی این مقاله در ابتدا به تهدیدها و آسیب‌پذیری‌های مؤلفه‌های اصلی جویش‌گرهای بومی پرداخته است؛ سپس در همین راستا الزامات و سیاست‌های امنیتی برای سه مؤلفهٔ اصلی خزش‌گر، نمایه‌سازی و رتبه‌بند که منجر به امن‌سازی معماری جویش‌گرهای بومی می‌شود، ارائه شده است.

واژگان کلیدی: جویش‌گر، خزش‌گر، نمایه‌ساز، رتبه‌بند و سئو منفی.

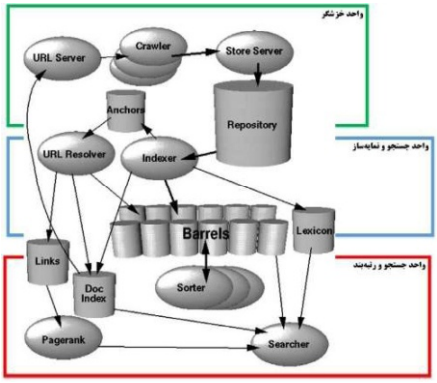
۱- مقدمه

امروزه با رشد شبکه‌های رایانه‌ای و پیشرفت در زمینه دستگاه‌هایی مثل تلفن‌های همراه هوشمند، تبلت‌ها، لپ‌تاپ‌ها و غیره تقاضا و جستجو برای محتواهای مبتنی بر وب افزایش یافته است. از طرفی با این رشد فزاینده در فناوری، تعداد حملات و آسیب‌پذیری‌ها نیز با همان ضریب افزایش پیدا کرده‌اند؛ به‌طوری‌که اثر و پیچیدگی حملات روی سامانه‌های قربانی نسبت به گذشته بیشتر شده است [1]؛ بنابراین ازآنجاکه جویش‌گرها تحت وب عمل می‌کنند، هر حمله تحت وب، تهدیدی برای آنها است. در ادامه ابتدا تهدیدهای مؤلفه‌های جویش‌گرها مشخص شده، سپس الزامات و سیاست‌های امنیتی که در جویش‌گرها باید وجود داشته باشد تا در برابر تهدیدها مقاوم باشند، تشریح شده است؛ زیرا وجود الزامات و سیاست‌های امنیتی برای مؤلفه‌های جویش‌گر، باعث امن‌شدن معماری جویش‌گر می‌شود.

روند کار جویش‌گرها بدین صورت است که وقتی

* نویسندهٔ عهده‌دار مکاتبات

جستجویی در یک جویش‌گر انجام و نتایج جستجو ارائه می‌شود، کاربران نتیجه کار بخش‌های متفاوت آن را می‌بینند. جویش‌گر از قبل پایگاه داده‌اش را آماده کرده است و این گونه نیست که درست در همان لحظهٔ جستجو، تمام وب را بررسی کند. گوگل و هیچ جویش‌گر دیگری توانایی انجام این کار را ندارند. همهٔ آن‌ها در زمان پاسخ‌گویی به جستجوهای کاربران، تنها در پایگاه داده‌ای که در اختیار دارند به جستجو می‌پردازند و نه در کل اینترنت. جویش‌گر به‌کمک بخش‌های متفاوت خود، اطلاعات مورد نیاز را از قبل جمع‌آوری، تجزیه و تحلیل و سپس آن را در پایگاه داده‌اش ذخیره و به هنگام جستجوی کاربران، تنها پایگاه داده خودش را بررسی می‌کند. مؤلفه‌های مجزا و اصلی یک جویش‌گر به چندین بخش تقسیم می‌شود که عبارتند از [2]، [3]، [4]: ربات و عنکبوت[1]، خزش‌گر، نمایه‌ساز، پایگاه داده و رتبه‌بند. به‌طورمعمول هر جویش‌گر از معماری متفاوتی برای انجام جستجوها و پاسخ به درخواست‌های کاربران استفاده می‌کند؛ اما در این قسمت، یک معماری کلی و عمومی که مؤلفه‌ها و بخش‌های آن در بیشتر جویشگرها وجود دارد، بررسی می‌شود. شکل (۱) چنین معماری‌ای را نشان می‌دهد (جویش‌گر گوگل نیز از این معماری استفاده می‌کند) [5].



(شکل-۱): معماری عمومی جویشگرها [5]

مهم‌ترین بخش‌هایی که در هر یک از سه مؤلفه خزش‌گر، نمایه‌ساز و رتبه‌بند تعریف می‌شوند، در ادامه تشریح شده است.

Repository و Store Server: خزش‌گر پس از دریافت فهرست URLها آنها را واکشی و نتیجه کار را به مؤلفه Store Server ارسال می‌کند. Store Server پس از فرآیند فشرده‌سازی صفحات، آنها را در Repository ذخیره می‌کند و برای هر صفحه وب که در Repository ذخیره می‌شود، یک شماره ID به نام DocID تخصیص داده می‌شود؛ (DocIDها در واحدهای[2] Barrels و Doc index ذخیره می‌شوند).

Sorter و Indexer: فرآیند نمایه‌سازی توسط مؤلفه‌های Sorter و Indexer انجام می‌شود. Indexer اسناد وب ذخیره‌شده در Repository را از حالت فشرده خارج و آنها را تجزیه می‌کند. هر سند به مجموعه‌ای از کلمات به نام Hits تبدیل می‌شود (رکوردهای Hits شامل اطلاعات اضافی در مورد اسناد و صفحات مثل رنگ، ظرفیت، قلم و غیره هستند). همچنین Indexer همه پیوندهای یک صفحه وب را تجزیه و اطلاعات مهم در مورد پیوندها را در واحد Anchors ذخیره می‌کند. Sorter تمام سندهای موجود در Barrels را بر اساس DocIDها مرتب و سپس عملیات نمایه معکوس را برای صفحات وب ایجاد می‌کند.

Barrels: این واحد پایگاه داده نمایه‌ساز محسوب می‌شود و نتایج حاصل از نمایه‌سازی را که به‌صورت رکوردهای Hits هستند، در خود ذخیره می‌کند.

Anchors: این واحد تمام پیوندها و اطلاعات مربوط به پیوندهای صفحات وب را به‌تفکیک مشخص و ذخیره می‌کند و نشان می‌دهد که هر پیوند از کجا سرچشمه گرفته یا به کجا اشاره می‌کند.

Lexicon: نمایه‌ساز پس از اینکه صفحات وب را تجزیه کرد، تمام کلمات (شامل کلمات کلیدی) و اطلاعاتی اضافی هر سند را در واحد Lexicon قرار می‌دهد. همچنین در این واحد یک جدول درهم‌سازی[3] نیز وجود دارد که هر رکورد آن به کلمات صفحات وب اشاره می‌کند.

URL Resolver: این واحد فایل‌ها را از Anchors دریافت می‌کند و پس از تجزیه، آنها را به واحدهای دیگر ارسال می‌کند.

Links: Links پایگاه داده‌ای است که تمام پیوندهای ورودی و خروجی یک صفحه وب را در خود ذخیره می‌کند. این پیوندها برای رتبه‌بندی صفحه وب به واحد Pagerank فرستاده می‌شود (زیرا پیوندها بیشترین تأثیر را در مؤلفه رتبه‌بند جویش‌گرها برای رتبه‌بندی صفحات وب دارند).

[1] Spider

[2] Module

[3] Hash Table

Doc index: این واحد ID تمام صفحات وب را نگهداری می‌کند. علاوه‌بر Doc index واحد Barrels نیز ID صفحات را نگهداری می‌کند (زیرا هر صفحه وب با یک شماره ID شناسایی می‌شود).

Pagerank: مهم‌ترین واحد جویش‌گر مؤلفه رتبه‌بند آن است. در Pagerank صفحات وب براساس پارامترهایی مثل پیوندها (پیوندهای ورودی و خروجی) رتبه‌بندی می‌شوند و خروجی این مرحله به سمت واحد Searcher فرستاده می‌شود.

Searcher: Searcher یا واحد جستجو، مؤلفه نهایی است که از اطلاعات سایر واحدها (Pagerank, Doc index, Lexicon) برای نمایش بهتر نتایج و پاسخ به درخواست‌های کاربران استفاده می‌کند.

با توجه به توضیحاتی که برای هر کدام از بخش‌های معماری جویش‌گرها ارائه شد، سه مؤلفه اصلی خزش‌گر (خزش‌گر از بخش‌های اصلی "سیاست‌های خزش، فاکتورهای طراحی، حملات تزریق کد، دسترس‌پذیری خزش‌گرها و پایگاه داده خزش‌گرها" تشکیل شده است)، نمایه‌ساز (بخش‌های اصلی نمایه‌ساز شامل "واحد جستجو، پایگاه داده نمایه‌ساز، سازوکارهای نمایه‌سازی، فاکتورهای طراحی" است) و رتبه‌بند (سیاست‌های رتبه‌بندی، سیاست های دفاعی در برابر بهینه‌سازی منفی جویش‌گرها یا سئو منفی[1] و وب‌گاه‌های آلوده[2]) تهدیدهایی دارند که درصورت نپرداختن به ضعف‌های امنیتی، عملکرد جویشگرها کاهش می‌یابد و درنهایت منجر به افشای اطلاعات غیر عمومی وب سایت‌ها، افشای اطلاعات خصوصی کاربران، خزش و نمایه‌شدن صفحات وب سایت‌های آلوده، بدخواه[3] و مخرب، ارائه نتایج غیر مرتبط و عدم محبوبیت جویش‌گر می‌شود. آمارها نشان می‌دهد که جویش‌گرهای بزرگی مثل گوگل در برابر برخی از تهدیدهای خزش‌گر و رتبه‌بند آسیب‌پذیر بوده‌اند [6]،[7]. طبق بررسی‌های صورت‌گرفته جویش‌گرهای بومی مثل پارسی‌جو کمبودهایی در زمینه سه مؤلفه اصلی دارند که تهدیدی برای آنها به شمار می‌رود؛ بنابراین استفاده از الزامات امنیتی ارائه‌شده در این مقاله به روند ارتقای جنبه‌های امنیتی جویش‌گرهای بومی کمک می‌کند.

[1] Negative Search Engine Optimization(SEO)
[2] Pollution Site
[3] Malicious

۲-تهدیـــــدها و آســـــیب‌پذیری‌های مؤلفه‌های جویش‌گرهای بومی

در این بخش تهدیدها و آسیب‌پذیری‌های سه مؤلفهٔ اصلی خزش‌گر، نمایه‌ساز و رتبه‌بند در جویش‌گرهای بومی بررسی شده است.

۲-۱- تهدیـــدها و آســـیب‌پذیری‌های مؤلفـــه خزش‌گر

خزش‌گرهای وب کار اصلی کاوش داده‌ها و اطلاعات را برعهده دارند و جویش‌گرها براساس داده‌های خزش‌شده به کاربران پاسخ می‌دهند. در ادامه برخی از انواع حملات، تهدیدها و آسیب‌پذیری‌هایی که به‌طور مستقیم یا غیر مستقیم روی خزش‌گرها تأثیر دارند، تشریح شده است [8].

۲-۱-۱- فاکتورهای طراحی خزش‌گر

پارامترهای طراحی و نیازمندی‌های مربوط به مؤلفه خزش‌گر، تأثیر زیادی در فرآیند پاسخ‌گویی به پرس‌وجوهای کاربران دارند. به‌طور مثال اگر جنبه‌های امنیتی برای مسائلی مانند پهنای باند، ساختمان داده، کنترل کننده‌ها، سامانهٔ پردازشی و منابع ذخیره‌سازی در خزش‌گرها لحاظ نشود، ممکن است، خزش‌گر در زمان مناسب وظایف خود را نتواند انجام دهد و عدم توجه به مسائلی از این قبیل، تهدیدی برای مؤلفه خزش‌گر محسوب می‌شوند [8]، [9].

۲-۱-۲- حملات منبع خدمت

به‌طورمعمول خزش‌گرها برنامه‌هایی هستند که پیوسته صفحات وب را بررسی می‌کنند و به‌محض اینکه وب‌سایت یا صفحه جدیدی ایجاد شد یا صفحات تغییر کردند، تغییرات را با انجام عملیاتی خاص، در قسمت نمایه‌سازی اعمال می‌کنند؛ بنابراین اگر این برنامه‌های کاربردی برای مدت زمان کوتاهی غیر فعال شوند، پاسخ به پرس‌وجوهای کاربران بر اساس صفحات و اطلاعات قدیمی انجام خواهد شد؛ علاوه‌براین ممکن است، طی این زمان صفحاتی آلوده ایجاد شوند؛ به‌همین دلیل لازم است با هر عاملی که باعث وقفه در کار خزش‌گر می‌شود، مقابله شود؛ چون مؤلفه‌های دیگر از خروجی خزش‌گر استفاده می‌کنند، وقفه در کار خزش‌گر باعث می‌شود مؤلفه‌های زیرین نیز تحت تأثیر قرار گیرند.

حملات منبع خدمت[1] یکی از تهدیدهایی هستند که منجر می‌شود جویشگرها نتوانند سرویس مناسب را در زمان تعیین شده ارائه کنند؛ زیرا زمانی که مؤلفه خزش‌گر تحت حمله منع خدمت است، ساختار کلی جویش‌گر تحت تأثیر قرار می‌گیرد و خدماتی که جویش‌گر ارائه می‌کند، براساس محتواها و داده‌هایی است که درقبل آنها را خزش کرده است؛ به‌عنوان مثال زمانی که یک کاربر سعی می‌کند جستجویی در مورد یک پایگاه داده که محتوای آن خیلی سریع به‌روزرسانی می‌شود، داشته باشد، جویش‌گر به‌اشتباه محتوایی را که از قبل پایگاه داده در خود موجود دارد، به کاربر نمایش می‌دهد.

۳-۱-۲- تزریق کد به ربات‌های خزش‌گر

امروزه نفوذگرها برای رسیدن به اهداف خود از ابزارها و ایده‌هایی استفاده می‌کنند که با کمترین ایجاد ردپا به بالاترین نتیجه دست پیدا می‌کنند؛ به‌طوری‌که آمار شرکت‌های امنیتی نشان می‌دهد، آنها برای حملات تزریق SQL از ربات‌های جویش‌گرها استفاده می‌کنند؛ زیرا با این عمل ربات‌های جویش‌گرها، حملات تزریق SQL را روی پایگاه‌های هدف انجام می‌دهند. با آلوده‌شدن ربات‌های جویش‌گر ممکن است، یک پایگاه از طرف یک جویش‌گر معتبر مثل گوگل مورد حمله قرار گیرد؛ در این شرایط اگر پایگاه مورد حمله، ربات‌های جویش‌گر را مسدود کند، آنگاه این عمل مانع نمایه‌سازی پایگاه خواهد شد و در غیر این‌صورت باید به ترافیک مخرب تولیدشده از سمت جویش‌گر اجازه فعالیت بدهد. درواقع هدف نفوذگرها از آلوده‌کردن ربات‌های جویش‌گرها، پاک‌کردن ردپای خود (گمنامی هکرها) و انجام عملیات خرابکارانه به‌صورت موفقیت‌آمیز روی وب‌سایت‌ها است. به‌عنوان مثال مهاجم به جای اینکه خود مستقیم به وب‌گاه site.com حمله کند، روی وب‌گاه خود پیوندی از حملات SQLI ایجاد و مابقی عملیات هک را به ربات‌های جویشگر واگذار می‌کند. بدین‌ترتیب ربات‌های جویش‌گرها با اجرای پیوند SQLI تولید شده توسط نفوذگر، وب‌گاه site.com را هدف حمله قرار می‌دهند [7].

۴-۱-۲- نقض حریم خصوصی وب‌گاه‌ها و سیاست‌های خزش

رفتار ربات‌ها و بایگانی‌کردن صفحات وب‌گاه توسط جویش‌گر را طبق پروتکل‌های پذیرفته‌شده زیر می‌توان کنترل کرد. این پروتکل‌ها عبارتند از [10]:

Robots.txt, XML Sitemap, Robots Meta Tag, Rel=Nofollow

پروتکل‌های XML Sitemap و Robots.txt برای کل وب‌گاه تعریف می‌شوند، به‌عبارتی سیاست‌های کلی وب‌گاه را تعریف می‌کنند. Robots Meta Tag برای هر صفحه وب به‌طور جداگانه لحاظ می‌شود و Rel=Nofollow به‌طور مجزا برای هر پیوند در نظر گرفته می‌شود. بیشتر جویش‌گرها این پروتکل‌ها را رعایت می‌کنند؛ اما هیچ‌کدام مجبور به رعایت آنها نیستند. درواقع این پروتکل‌ها مواردی را به جویش‌گرها توصیه می‌کنند و رعایت و عدم رعایت اینها به جویش‌گرها بستگی دارد؛ زیرا جویش‌گرها به‌خودی خود جسور و اگر مؤدب[2] باشند (ربات‌ها، اسپایدرها و خزش‌گرها) برای حفظ حریم خصوصی وب‌گاه‌ها، این توصیه‌ها را رعایت می‌کنند. بهتر است که جویش‌گرها، توصیه تمامی وب‌گاه‌هایی را که از پروتکل‌های بالا استفاده کرده‌اند، رعایت کنند؛ زیرا مالکان وب‌گاه‌ها ممکن است اطلاعاتی را روی وب‌گاه قرار دهند که برای حفظ حریم خصوصی، تمایلی برای خزش موقت آن صفحات نداشته باشند. به‌عنوان مثال اگر نشانی یک وب‌گاه تغییر کرده است و مالک وب‌گاه تمایلی ندارد که جویش‌گرها صفحات قدیمی را خزش کنند، این تغییرات را در XML Sitemap و Robots.txt اعمال می‌کند. به‌همین دلیل جویش‌گرها برای اعمال تغییرات وب‌گاه‌ها و عدم خزش صفحات منع‌شده برای حفظ حریم خصوصی وب‌گاه‌ها، بهتر است که توصیه‌های پروتکل‌های بالا را رعایت کنند. اگر جویش‌گرها توصیه‌های پروتکل‌های بالا را رعایت نکنند، ممکن است مهاجمان از صفحاتی که به‌طور مؤدب خزش نشده‌اند، سوء استفاده کنند، به‌عنوان مثال دستیابی مهاجمان به صفحات پاک‌شده یک وب‌گاه که حاوی اطلاعات مهم است. به‌طورکلی اگر خزش‌گرها توصیه‌های وب‌گاه‌ها را رعایت نکنند یا اگر سیاست‌های خزش مناسب برای خزش صفحات وب نداشته باشند، انواع تهدیدها مثل تله‌اسپایدر و حلقه بی‌نهایت برای خزش‌گرها اتفاق می‌افتد [8].

برخی از سیاست‌های کلی و عمومی خزش‌گرها در مواجه با وب‌گاه‌ها عبارتند از [10]، [11]: ۱- سیاست

[1] Denial-Of-Service (DOS)

[2] Politeness

انتخاب[1]: این سیاست به‌طورکلی وضعیت دانلود صفحات وب را مشخص می‌کند. ۲- سیاست بازمشاهده[2]: این سیاست تغییرات صفحات وب را کنترل می‌کند. ۳- سیاست ادب[3]: این سیاست برای جلوگیری از بارگذاری دوباره صفحات، اعمال‌کردن تغییرات وب‌گاه‌ها و حفظ حریم خصوصی آنها استفاده می‌شود. ۴- سیاست موازی‌سازی[4]: این سیاست نشان می‌دهد که خزش‌گرهای توزیع‌شده چگونه باید با یکدگیر هماهنگ شوند.

۲-۲- تهدیدها و آسیب‌پذیری‌های مؤلفه نمایه‌ساز

مؤلفه نمایه‌ساز وظیفه جمع‌آوری، ذخیره‌سازی و تجزیه داده‌ها را برای بازیابی درست و آسان اطلاعات بر عهده دارد. بدون نمایه‌سازی جویش‌گرها باید زمان زیادی را صرف جستجو در پایگاه داده خود کنند.

۲-۲-۱- تهدیدهای واحد جستجو

وظیفه اصلی واحد جستجو پاسخ به درخواست‌ها و پرس‌وجوهای کاربران است و این واحد موظف است درخواست‌های جستجو را از کاربران دریافت و آنها را در قالبی مناسب جهت شروع فرآیند جستجو سازماندهی کند. جویش‌گرها به‌طورمعمول برای واحد جستجو می‌توانند از هر بستری مثل آپاچی سُلر[5]، ElasticSearch، Compass، DocFetcher، Swiftype، Ferret و غیره (این سکوهای نرم‌افزاری روی نمایه‌ساز آپاچی لوسین پیاده‌سازی می‌شوند) استفاده کنند. آپاچی سُلر و ElasticSearch برای پرس‌وجوهای زیاد در مقیاس بزرگ عملکرد بسیار خوبی دارند؛ اما هر کدام از این دو مورد آسیب‌پذیری‌هایی دارند که در صورت سوء استفاده از آنها مهاجم به‌راحتی به کل پایگاه داده از راه دور دسترسی می‌تواند داشته باشد. برخی از آسیب‌پذیری‌های واحد جستجو عبارتند از [9]، [12]، [13]:
Cross-site request forgery (CSRF), Directory traversal (Dir.Trav), Cross-site scripting (XSS), Exec Code Inject

طبق بررسی‌های صورت‌گرفته آپاچی سُلر در بیشتر جویش‌گرها از جمله جویش‌گرهای بومی استفاده می‌شود و مهمترین آسیب‌پذیری آپاچی سُلر مربوط به حمله پیمایش فهرست[6] است. سناریوی حملات پیمایش فهرست بدین صورت است که خدمت‌گزارهای وب نیز مانند ساختار فایلی که در سیستم‌عامل وجود دارد، دارای سلسله‌مراتب هستند، خدمت‌گزارهای وب به‌طورمعمول دسترسی به ریشه فهرست‌های خود را محدود می‌کنند؛ کاربران در این حالت به فهرست‌های زیرمجموعه می‌توانند دسترسی پیدا کنند؛ اما به مرحله بالاتر که ریشه است نمی‌توانند دسترسی پیدا کنند؛ همچنین نمی‌توانند به‌صورت موازی از پوشه‌های کنار دستی نیز استفاده کنند و تنها به محلی ارجاع داده می‌شوند که به آن دسترسی دارند. برخی اوقات به‌علل مختلف که بارزترین آن‌ها سهل‌انگاری مدیر خدمت‌گزار وب است، دسترسی‌های مناسب برای پوشه ریشه در نظر گرفته نمی‌شود و همین امر می‌تواند باعث شود مهاجم به فهرست ریشه خدمت‌گزار وب دسترسی پیدا کند. پیمایش فهرست سوءاستفاده از همین نقاط ضعفی است که در سطوح دسترسی یا عدم اعتبارسنجی داده‌های ورودی کاربر به‌ویژه نام فایل‌ها در خدمت‌گزار وب است که درنهایت می‌تواند باعث شود مهاجم با واردکردن نویسه‌های مشخصی به فهرست ریشه[7] دسترسی پیدا کند. هدف اینگونه حملات دسترسی پیداکردن به پوشه‌هایی است که نبایستی به‌صورت عمومی در دسترس قرار بگیرند، در این حمله درنهایت نفوذگر می‌تواند فایل‌های موجود در قسمت محدودشده[8] را دست‌کاری کند و نتیجه لازم را بگیرد.

۲-۲-۲- فاکتورهای طراحی نمایه‌ساز و موازی‌سازی فرآیندها

یکی از مهم‌ترین مسائلی که باید در واحد نمایه‌ساز جویش‌گرها در نظر گرفت، فاکتورهای طراحی است؛ زیرا این فاکتورها عامل اصلی سرعت و عملکرد نمایه‌ساز در قبال پرس‌وجوهای کاربران است. برخی از مهم‌ترین فاکتورها و پارامترهای اصلی که در طراحی معماری نمایه‌ساز جویش‌گر استفاده می‌شود، عبارتند از: فاکتورهای ادغام، فاکتورهای ساختمان داده‌ای، اندازه نمایه‌سازی، روش‌های ذخیره‌سازی،

[1] Selection Policy
[2] Re-Visit Policy
[3] Politeness Policy
[4] Parallelization Policy
[5] Apcache Solr

[6] Directory Traversal
[7] Parent Directory
[8] Restricted

تحمل‌پذیری خطا، سرعت جستجو و غیره [9].

یکی از چالش‌های اساسی دیگری که جویش‌گرها با آن مواجه هستند، مدیریت فرآیندهای پردازش سریالی است. به‌عنوتن مثال وقتی یک سند جدید به مجموعه اسناد اضافه می‌شود، نمایه‌ساز باید به‌روزرسانی شود؛ اما نمایه‌ساز در آن زمان نیاز دارد تا به پرس‌وجوهای کاربران نیز پاسخ دهد. بنابراین اینجا دو وظیفه رقابتی (وظیفه به‌روزرسانی و وظیفه پاسخ به پرس‌وجوها) اتفاق می‌افتد که نمایه‌ساز باید آنها را مدیریت کند. این چالش در ذخیره‌سازی‌ها و پردازش‌های توزیع‌شده بیشتر آشکار می‌شود.

۳-۲-۲- زمان‌بندی و دوره به‌روزرسانی نمایه‌ساز

پاسخ به درخواست‌های کاربران بر اساس نمایه‌سازی‌های به‌روزشده یک امر مهم در جستجو است؛ به‌طوری که باید همیشه سرویس‌ها بر اساس اطلاعات تازه و به‌روز ارائه شود. علاوه‌بر مسئله تازگی صفحات وب، به این نکته مهم باید توجه داشت که اگر یک صفحه وب در زمان T نمایه شده باشد و مهاجم در بازه زمانی T تا T+n صفحه را تغییر دهد و یک صفحه آلوده ایجاد کند و جویشگر تغییراتی را که در بازه زمانی T تا T+n اتفاق می‌افتد، ندیده باشد، آنگاه ممکن است کاربران به صفحه‌ای که مهاجم در بازه زمانی T تا T+n (صفحه آلوده) ایجاد کرده است، هدایت شوند. بنابراین اگر دوره به‌روزرسانی نمایه‌سازی صفحات زیاد باشد، آنگاه جویش‌گر نمی‌تواند صفحه آلوده و تغییرکرده را شناسایی کند.

۴-۲-۲- سئوی منفی

اگر چه سئوی منفی بیشتر مؤلفه رتبه‌بند جویشگرها را تهدید می‌کنند؛ اما یک جویش‌گر اگر عکس‌العمل خوبی در برابر سئوی منفی نداشته باشد، آنگاه مؤلفه نمایه‌ساز جویشگر، نیز تحت تأثیر این حمله قرار می‌گیرد. به‌عنوان مثال مهاجم با استفاده از سئوی منفی، وب‌گاه رقیب خود (قربانی) را هدف قرار می‌دهد؛ سپس جویش‌گر در این مرحله درصورتی‌که راه‌کار مناسبی در زمینه سئوی منفی نداشته باشد، ابتدا رتبه وب‌گاه قربانی را کاهش می‌دهد و به‌دلیل کاهش رتبه درنهایت برخی از صفحات وب‌گاه قربانی را ممکن است از سامانهٔ نمایه‌سازی خود حذف کند، به‌همین دلیل جویش‌گرهای بزرگی مثل گوگل در زمینه مقابله با سئوی منفی خیلی خوب عمل کرده‌اند و امروزه سیاست‌های دفاعی آنها به‌صورت دستی و خودکار[1] در فرایند جستجو استفاده می‌شوند. سایر تهدیدهای سئوی منفی و الزامات امنیتی آن که گوگل نیز از آنها بهره می‌برد، در قسمت مؤلفه رتبه‌بند تشریح شده است.

۵-۲-۲- امنیت پایگاه داده‌ها

پایگاه داده یکی از واحدهای مهم برای مؤلفه نمایه‌ساز به‌شمار می‌آید؛ زیرا مؤلفه نمایه‌ساز تمام اطلاعاتی را که از بخش خزش‌گر دریافت می‌کند، نمایه‌سازی و درنهایت خروجی حاصل از نمایه‌سازی را در پایگاه داده قرار می‌دهد. برخی از تهدیدهای امنیتی پایگاه داده‌ها که شامل جویش‌گرها نیز می‌شود، عبارتند از: سوءاستفاده از دسترسی‌های اضافی، سوءاستفاده از دسترسی‌های قانونی، ترفیع حق دسترسی، آسیب‌پذیری‌های پلتفرم، آسیب‌پذیری‌های استنتاجی پایگاه داده، تزریق کد، فعال‌بودن ویژگی‌های غیرضروری DBMS، سرریزی بافر، عدم رمزنگاری داده‌ها و نبود اطلاعات پشتیبان.

۳-۲- تهدیدها و آسیب‌پذیری‌های مؤلفه رتبه‌بند

بهینه‌سازی جویش‌گرها یکی از مهم‌ترین روش‌هایی است که مالکان وب‌گاه‌ها برای افزایش رتبه و بهبود حضور وب‌گاه از آن استفاه می‌کنند. درصورت استفاده درست از سئو رتبه وب‌گاه‌ها در بخش رتبه‌دهی جویشگرها افزایش در غیر این صورت، سئو اثر معکوس روی رتبه‌دهی وب‌گاه‌ها دارد. حملات مربوط به سامانهٔ رتبه‌بندی جویش‌گرها بیشتر با نام سئوی منفی شناخته می‌شوند و هدف این حملات تغییر سامانهٔ رتبه‌دهی و رتبه‌بندی جویش‌گرها است. برخی از روش‌های سئوی منفی که در [14]، [15] به‌عنوان سئوی کلاه‌سیاه[2] شناخته می‌شوند، در ادامه تشریح شده است.

۱-۳-۲- حمله ربودن پراکسی یا هک پراکسی

روش کار حمله ربودن پراکسی بدین صورت است که ابتدا مهاجم یک رونوشت از صفحات وب‌گاه هدف (وب‌گاه قربانی) را روی یک خدمت‌گزار پراکسی ایجاد و سپس با استفاده از روش‌های دیگر سئوی کلاه‌سیاه و پیوندهای کمکی، پیوندی

[1] Automatic
[2] Black Had SEO

از وب‌گاه‌های معتبر دیگر به صفحات موجود در خدمت‌گزار پراکسی ایجاد می‌کند و این عمل در درازمدت منجر به افزایش رتبه صفحات موجود در خدمت‌گزار پراکسی می‌شود. بنابراین جویش‌گرها وقتی دو وب‌گاه با محتوای یکسان را ملاقات می‌کنند، رتبه دو وب‌گاه را در سامانهٔ رتبه‌بندی تغییر می‌دهند؛ زیرا جویش‌گرها ممکن است به‌دلیل تکراری‌بودن صفحات، صفحات وب‌گاه معتبر را از سامانهٔ شاخص‌دهی خود حذف یا رتبه پایین‌تری به صفحات وب‌گاه معتبر تخصیص دهند. اگر چنین اتفاقی برای وب‌گاه معتبر بیفتد، مهاجم به هدف خود که همان کاهش رتبه صفحات وب‌گاه معتبر بوده، رسیده است (به‌عنوان مثال نشانی وب‌گاه معتبر می‌تواند www.example.com و نشانی وب‌گاه جعلی که روی خدمت‌گزار پراکسی قرار گرفته www.example.net باشد). هدف از حمله ربودن پراکسی[1] تغییر صفحات یک وب‌گاه معتبر به صفحات غیر معتبر است؛ زیرا با این عمل رتبه صفحات یک وب‌گاه در نتایج و مؤلفه رتبه‌بند و نمایه‌ساز جویش‌گر تغییر می‌کند. همچنین یک مهاجم ممکن است با ربودن پراکسی به یکسری اطلاعات مهم از وب‌گاه رقیب دسترسی پیدا کند، یا درنهایت درخواست کاربران وب‌گاه رقیب را به صفحات جعلی و آلوده (صفحات شامل بدافزار، تروجان و غیره) هدایت کند.

۲-۳-۲- چاشنی کلمهٔ کلیدی

چاشنی کلمه کلیدی[2] یک روش غیر مجاز در سئو است و باعث می‌شود که یک وب‌گاه به‌طور موقت یا دائمی از نتایج جویش‌گرها حذف شود. چاشنی کلمه کلیدی زمانی اتفاق می‌افتد که تعداد، چگالی و تنوع کلمات کلیدی موجود در متا تگ‌ها یا صفحات یک وب‌گاه بیش از حد زیاد باشد، از این طریق سعی می‌شود که رتبه وب‌گاه را به‌طور مصنوعی افزایش دهد. کلمات کلیدی که در چاشنی کلمه کلیدی استفاده می‌شوند، بیشتر به‌طور مخفی در کدهایی HTML مربوط به قلم، رنگ، عکس و غیره قرار می‌گیرند. چاشنی کلمه کلیدی به‌همراه حملات دیگر مثل ربودن پراکسی نیز استفاده می‌شود. بر اساس بررسی‌های صورت‌گرفته در منبع [۳۳] جویش‌گر گوگل هر وب‌گاهی را که از چاشنی کلمه کلیدی استفاده کند، شناسایی و رتبه آن را کاهش می‌دهد؛ اما جویش‌گرهای دیگر از جمله جویش‌گرهای بومی سازوکار خاصی در این زمینه ندارند.

۳-۳-۲- مزرعه پیوند

مزرعه پیوند[3] یکی دیگر از روش‌هایی است که مالکان وب‌گاه برای دست‌کاری سامانهٔ رتبه‌بندی جویش‌گرها استفاده می‌کنند. درواقع مزرعه پیوند به گروهی از وب‌گاه‌ها گفته می‌شود که بدون اینکه ارتباط مفهومی با هم داشته باشند، دست به مبادله گسترده پیوند می‌زنند تا محبوبیت خود را افزایش دهند. در نگاه نخست مزرعه پیوند یک پیوند معمولی از یک وب‌گاه به نظر می‌رسد؛ درحالی‌که اینگونه نیست و در بیش‌تر موارد به‌طور تصادفی، ناخواسته یا هدف‌دار از وب‌گاه‌های دیگر سرچشمه می‌گیرد. درواقع اگر درجه یا چگالی پیوندهای ورودی یا خروجی یک وب‌گاه از یک مقدار آستانه بیشتر باشد یا پیوندهای ورودی از وب‌گاه‌هایی سرچشمه بگیرند که رتبه پایین و محتوای نامناسب دارند (مثل Bad Links & Bad Neighbours, Link Spamming)، سامانهٔ رتبه‌بندی جویش‌گرها برای وب‌گاه مقصد تغییر می‌کند. امروزه جویش‌گرها براساس یکسری ویژگی‌های خاص، صفحات وبی را که با مزرعه پیوند مرتبط هستند، شناسایی می‌کنند و در نتایج جستجو نشان نمی‌دهند. البته در برخی از موارد ممکن است، فقط رتبه وب‌گاه کاهش یابد (برخی از استراتژی‌های سئوی موجود در جویش‌گرها، برای مقابله با مزرعه پیوند استفاده می‌شود).

۴-۳-۲- ریسندگی محتوا

ریسندگی محتوا[4] یکی دیگر از روش‌هایی است که برای بهینه‌سازی جویش‌گرها استفاده و اما استفاده نامناسب از آن باعث تغییر رتبه وب‌گاه‌ها، در سامانهٔ رتبه‌بندی جویش‌گرها می‌شود. ریسندگی محتوا فرایند بازنویسی یک محتوا (عبارت، صفحه وب و غیره) از روی یک محتوای اصلی است؛ به‌گونه‌ای که رونوشت ایجادشده باعث ایجاد محتوای تکراری نشود. درواقع هدف از ریسندگی محتوا ایجاد یکصد یا هزار محتوای ایجادشدهٔ جدید از روی محتوای اصلی، با نسخه‌های گوناگون و در زمان‌های متفاوت است؛ زیرا درصورت تکراری‌شدن محتوای صفحات وب به‌وسیلهٔ ریسندگی محتوا، حمله‌ای مانند ربودن پراکسی برای وب‌گاه اصلی اتفاق

[1] Proxy Hijacking (or Proxy Hacking)
[2] Keyword stuffing
[3] Link Farm
[4] Article spinning

می‌افتد؛ درواقع اگر ریسندگی محتوا به‌طور پیوسته انجام شود، درنهایت منجر به حمله ربودن پراکسی خواهد شد.

۵-۳-۲- صفحه پنهان

صفحه پنهان[1] روشی در بهینه‌سازی جویش‌گرها است؛ که در آن محتوای ارائه‌شده به اسپایدر یا ربات جویش‌گر، با چیزی که کاربر در مرورگر خود مشاهده می‌کند، متفاوت است. صفحه‌ای که به ربات جویش‌گر نشان داده می‌شود، ممکن است از لحاظ چگالی کلمات کلیدی، پیوندهای ورودی و خروجی، نوع محتوا و غیره مناسب باشد؛ اما صفحه‌ای که به کاربر نشان داده می‌شود، ممکن است یک صفحه هرز آگهی، غیر اخلاقی و بی‌ارزش باشد. بنابراین با حمله صفحه پنهان، صفحات هرزنامه در رتبه‌بند جویشگر دستخوش تغییر نمی‌شود و فقط صفحهٔ نشان داده شده به ربات‌ها با افزایش یا کاهش رتبه مواجه می‌شود. درواقع وب‌گاه‌ها برای ارائه محتواهای متفاوت به کاربران و جویشگرها از نشانی IP یا header User-Agent HTTP درخواست‌ها استفاده می‌کنند.

۶-۳-۲- صفحه درگاه

صفحات درگاه[2] به‌طورمعمول شامل صفحاتی هستند که هر کدام برای یک کلیدواژه یا عبارت خاص بهینه‌سازی شده است و هیچگونه ارزش محتوایی ندارند. در بسیاری از موارد به‌خصوص برای بالابردن رتبه صفحات یا عبارات خاص در جویش‌گرها طراحی شده‌اند و درنهایت کاربران را به یک سایت از قبل مشخص‌شده، هدایت می‌کنند. صفحات درگاه برای کاربران بسیار بد است؛ زیرا باعث می‌شود که کاربران به صفحات مشابه متعددی که صفحه مقصد آنها یکسان است، هدایت شوند. به این معنی که اگر کاربر روی هر پیوند موجود در صفحه درگاه کلیک کند، مقصد همه پیوندها یک صفحهٔ از قبل تعیین‌شده است. این صفحات چه بر مبنای یک دامنه و چه چند دامنه باشد، تنها خستگی را برای کاربر به همراه می‌آورند.

۷-۳-۲- وب‌گاه آلوده[3]

موقعی که یک وب‌سایت توسط برنامه‌های مخرب مثل ویروس‌ها و تروجان‌ها آلوده می‌شود، خزش‌گر جویش‌گر که به احتمال بالایی آن صفحات وب را کشف کرده، ممکن است آنها را در نتایج جستجو نشان دهد. در چنین مواردی یک جویش‌گر ایمن به جای اینکه مستقیم صفحات آلوده را به کاربران نشان دهد، آن را با پیامی مبنی بر اینکه "این سایت یا صفحه وب ممکن است، برای سامانهٔ شما خطرناک باشد" به کاربران هشدار می‌دهد و کاربران با اختیار خود به آن صفحات رجوع می‌کنند.

۸-۳-۲- امنیت وب‌گاه

امروزه جویش‌گرها وب‌گاه‌هایی را که مورد خزش قرار می‌دهند از لحاظ امنیتی وارسی و نتیجه آن را در سامانهٔ رتبه‌بندی اعمال می‌کنند. بنابراین اگر یک وب‌سایت کمینه امنیت لازم را در برابر انواع حملاتی مانند نشت اطلاعات، SQLI, XSS و غیره نداشته باشد، رتبه آن وب‌گاه در سامانهٔ رتبه‌بندی جویش‌گرها تغییر خواهد کرد (امنیت وب‌گاه‌ها توسط ابزارهای خاص جویشگر ارزیابی می‌شود)، به‌طوری که جویش‌گر گوگل در سال ۲۰۱۴ اعلام کرد در زمینه امنیت وب‌سایت‌ها، سیاست‌های جدیدی را در سامانهٔ رتبه‌بندی خود اعمال می‌کند. بنابراین وجود سیاست‌های امنیتی برای ارزیابی سطح امنیت وب‌گاه‌ها برای اعتبار و محبوبیت جویش‌گرها یک امر الزامی است.

۳- الزامات و سیاست‌های امنیتی برای مؤلفه‌های اصلی جویش‌گرهای بومی

در این بخش تمام سیاست‌ها و الزامات امنیتی که جویش‌گرها باید برای سه مؤلفه اصلی خود یعنی خزش‌گر، نمایه‌ساز و رتبه‌بند در نظر بگیرند، تشریح شده است؛ زیرا درصورتی‌که جویش‌گرها سیاست‌های امنیتی کافی را در مؤلفه‌های اصلی خود نداشته باشند، ممکن است حملات مختلفی مثل سئوی منفی، حملات منع خدمت و حملات تزریق کد، جویش‌گرها را تهدید کند و این تهدیدها درنهایت باعث نمایش نتایج غیر مرتبط، Crashکردن جویش‌گرها و عدم محبوبیت آنها می‌شود.

۱-۳- تدوین الزامات و سیاست‌های امنیتی خزش‌گر

جویش‌گرها به‌طورمعمول بر اساس اطلاعاتی که خزش‌گرها

[1] Cloacking Page
[2] Doorway pages
[3] Pollution Web Site

جمع‌آوری می‌کنند، به درخواست‌های کاربران پاسخ می‌دهند. بنابراین مؤلفه خزش‌گر با خزش درست و بهینه وب، اطلاعات و داده‌های بهتری در اختیار سایر مؤلفه‌های جویش‌گر می‌تواند قرار دهد. در ادامه برخی از الزامات امنیتی کلی که باعث امن‌شدن خزش‌گرها می‌شود، ارائه شده است.

۱-۱-۳- الزامات امنیتی حملات تزریق کد

برای جلوگیری از حملات تزریق کد به خزش‌گرها، جویش‌گرها باید یکسری سیاست در الگوریتم‌های خزش اعمال کنند تا پیوندها قبل از خزش وارسی شوند. درواقع جویش‌گرها بهتر است، ابتدا پیوند اصلی را خزش و سپس ادامه پیوند را در صورتی خزش کنند که هیچ نویسه و کلمه کنترلی در آن وجود نداشته باشد. نحوه تزریق یک کد بدین صورت است که ابتدا مهاجم مانند حالت زیر، پیوندی را در وب‌گاه هدف قرار می‌دهد و خزش‌گر وقتی به چنین پیوندی برخورد کرد، سعی می‌کند محتوای داخل این پیوند را خزش کند؛ اما رجوع به پیوند زیر باعث اجرای یک دستور SQL روی پایگاه داده وب‌گاه www.example.com می‌شود (این یک نمونه ساده از حمله تزریق کد بود، حالت‌های پیشرفته‌تری نیز از این حمله وجود دارد که مهاجمان از آنها می‌توانند استفاده کنند).

http://www.example.com/index.php?id=number+union+delete+from+table_name...

برای وارسی پیوندها از دو رویکرد فهرست سیاه (یا سازوکار رهایی[1]) و فهرست سفید می‌توان استفاده کرد.

در روش فهرست سیاه ابتدا فهرستی از نویسه‌هایی که باید ترجمه شوند، ایجاد می‌شود و در فهرست سیاه قرار می‌گیرند؛ سپس ورودی‌های خزش‌شده با این فهرست سیاه مقایسه می‌شوند. حال اگر در داده‌های ورودی نویسه‌ای وجود داشته باشد که در فهرست سیاه هم موجود باشد، آنگاه آن نویسه به یک نویسهٔ بی‌معنی تبدیل می‌شود یا ممکن است آن نویسه حذف و ادامه پیوند خزش شود. استفاده از این روش خطراتی دارد؛ چون ممکن است، بعضی از نویسه‌های کنترلی در فهرست سیاه موجود نباشند؛ همچنین این روش به‌دلیل مقایسهٔ همهٔ ورودی‌ها با فهرست سیاه پیچیدگی زمانی‌اش زیاد است.

یکی دیگر از سازوکارهای دفاعی قابل قبول برای مسئله وارسی پیوندها، استفاده از فهرست سفید است. در این روش ابتدا یک ساختار اعتبارسنجی قوی پیاده‌سازی می‌شود که ورودی‌ها و پیوندهای قانونی و معتبر را می‌پذیرد.

۲-۱-۳- الزامات امنیتی فاکتورهای طراحی

پارامترهای طراحی و نیازمندی‌های مربوط به مؤلفه خزش‌گر تأثیر زیادی در فرآیند پاسخ‌گویی به پرس‌وجوهای کاربران دارند و عدم توجه به مسائلی از این قبیل، تهدیدی برای مؤلفه خزش‌گر محسوب می‌شوند. برخی از فاکتورهای امنیتی مورد نیاز در مؤلفه خزش‌گر عبارتند از:

- حل‌کنندهٔ DNS: استفاده از حل‌کنندهٔ DNS آسنکرون چندنخی
- خدمت‌گزارهایی با منابع مناسب: استفاده از لوح‌های سخت سریع مثل SSD، اسکازی، ساس[2] با حجم ذخیره‌سازی بالا
- پهنای باند بالا و کافی برای خزش بهتر صفحات وب
- مدیر یا کنترل‌کننده‌ها: اعمال سیاست‌های ملاقات وب‌گاه‌ها ، حذف و به‌روزرسانی سیاست‌های خزش
- الگوریتم‌های خزش: استفاده از الگوریتم‌های جستجوی عرضی

۳-۱-۳- مقابله با حملات منع خدمت

حملات منع خدمت/توزیع‌شده یکی از عواملی هستند که منجر می‌شود خزش‌گرها یا سامانهٔ جویش‌گر نتواند سرویس مناسب را در زمان تعیین‌شده ارائه کند؛ بنابراین داشتن راه‌کارهای لازم برای رفع چنین حملاتی یکی از الزامات امنیتی اساسی برای جویش‌گر محسوب می‌شود. برای جلوگیری از حملات منع خدمت در سطح خزش‌گر نیاز است که از نرم‌افزارهای امنیتی، مانند دیوار آتش‌های معمولی یا هوشمند (Web Application Firewall و Network Layer Firewall) استفاده شود. بهره‌گیری از دیوار آتش‌های مبتنی بر مدل سند شی[3] توصیه می‌شود. این دیوار آتش‌ها علاوه‌بر کشف حملات جدید سربار کمتری را نیز به شبکه وارد می‌کند.

۴-۱-۳- کاربردهای توصیفی امنیتی خزش‌گرها

برخی از کاربردهای امنیتی خزش‌گرها را در موارد زیر

[1] Escaping

[2] Serial Attached SCSI (SAS)

[3] Document Object Model (DOM)

می‌توان خلاصه و از نتایج آنها در سطوح نمایه‌ساز و رتبه‌بند استفاده کرد.

- کشف صفحات و وب‌گاه‌های آلوده درهنگام خزش وب؛
- کشف و مقابله با سئوی منفی در هنگام خزش وب؛
- ارائه راه‌کارهای امنیتی به وب‌گاه‌هایی که در سطوح پایین امنیت قرار دارند.

۵-۱-۳- سیاست‌های امنیتی خزش

همان‌طورکه در قبل ذکر شد، خزش‌گر از چند سیاست اصلی برای خزش محتوای وب‌گاه‌ها استفاده می‌کند که در ادامه نحوه بهره‌گیری مناسب این سیاست‌ها به‌طور کامل تشریح شده‌اند.

سیاست انتخاب: اولویت‌بندی و انتخاب صفحات مهم یک ضرورت محسوب می‌شود. بایستی صفحات وب بر اساس یک معیار اولویت‌بندی شوند و خزش‌گر وب بر این اساس حرکت کند تا مؤثرترین جمع‌آوری در کم‌ترین زمان صورت گیرد. استراتژی‌های متفاوتی برای انتخاب محتواها توسط خزش‌گرها وجود دارد که در بخش استراتژی‌های خزش خزش‌گرها ذکر خواهد شد [11].

سیاست بازمشاهده: از دید یک جویش‌گر عدم شناسایی رویدادها که منجر به داشتن رونوشت منسوخ‌شده از منابع می‌شود، هزینه دارد. تازگی و سن به‌طورمعمول جزء پراستفاده‌ترین توابع هزینه برای مقابله با رونوشت‌های قدیمی هستند. دو معیار تازگی و سن بدین منظور استفاده می‌شود تا بتوان تعیین کرد چه صفحاتی نیاز به مشاهده‌ی مجدد دارند. بنابراین زمان بازدید دوباره صفحات برای خزش‌گرهای یک جویشگر باید به‌گونه‌ای باشد که پاسخ به پرس‌وجوهای کاربران بر مبنای اطلاعات به‌روزشده انجام شود. سه رویدادی که ممکن است رخ دهد و باعث تغییر صفحات وب شود عبارتند از [11]: ۱- ایجاد یک صفحهٔ جدید که نیازمند اضافه‌شدن پیوندی جدید است. ۲- تغییر و به‌روزرسانی صفحات که ممکن است، تغییر جزئی یا بزرگ باشد. ۳- حذف یک صفحه که باعث می‌شود، خروجی خزش‌گر پیوندی را نگه‌داری کند که دیگر وجود ندارد.

سیاست موازی: خزش‌گر موازی خزش‌گری است که چندین فرآیند را به‌صورت موازی اجرا می‌کند. هدف آن بیشینه‌کردن نرخ بارگیری است؛ درحالی‌که سربار موازی‌سازی را کمینه و از بارگیری صفحات تکراری اجتناب می‌کند. جهت اجتناب بارگیری بیش از یک‌بار صفحات، سامانه‌های خزش‌گر، نیازمند سیاستی هستند که نشانی‌های جدید کشف‌شده طی خزش را انتساب دهند. در این صورت، اگر یک نشانی یکسان توسط دو فرآیند خزش پیدا شود، مشخص خواهد شد و از دوباره‌کاری جلوگیری می‌شود؛ زیرا با وجود سیاست‌های موازی جنبه امنیتی دسترس‌پذیری خدمت‌گزارهای خزش‌گر نیز بیشتر خواهد شد [11].

سیاست ادب: اگر یک خزش‌گر چندین درخواست در ثانیه انجام دهد و یا فایل‌های بزرگی را بارگیری کند، سرویس‌دهنده[1] زمان زیادی را صرف پاسخ‌گویی به درخواست‌های خزش‌گر خواهد کرد و درنهایت این کار باعث می‌شود که کارایی سرویس‌دهنده کمی تغییر کند. بنابراین استفاده از خزش‌گرها برای کاربردهایی مفید است؛ ولی هزینه‌هایی هم به‌دنبال دارد. اگر چه وب‌گاه‌ها برای ارائه و نمایش خدمات خود به مشتریان، باید به درخواست‌های خزش‌گر جویش‌گرها پاسخ دهند، به‌طوری‌که بیشتر مالکان وب‌گاه‌ها از روش‌هایی استفاده می‌کنند تا خزش‌گرهای جویش‌گرها وب‌گاه آنها را بیشتر ملاقات کنند؛ اما به‌طورکلی خزش‌گرها (خزش‌گرهای وب که شامل خزش‌گر جویش‌گرها است) باعث اعمال برخی از هزینه‌ها مثل اتلاف منابع شبکه، سربار پردازشی و غیره به خدمات‌دهنده‌ها می‌شوند [11]. یک راه حل برای رفع بخشی از هزینه‌های تحمیلی به خدمات‌دهنده‌ها، استفاده از پروتکل ممانعت از ربات‌ها است، این پروتکل با نام Robots.txt شناخته می‌شود. در همین اواخر، جویش‌گرهای تجاری بزرگ مانند Google، Ask Jeeves، MSN و Yahoo! Search از یک پارامتر اضافی به نام Crawl-Delay در فایل Robots.txt می‌توانند استفاده کنند که این پارامتر تعداد ثانیه‌های تأخیر بین درخواست‌های خزش‌گرها از یک سایت را نشان می‌دهد؛ زیرا اگر زمان بین درخواست‌ها خیلی کم باشد، آنگاه خزش‌گر یک جویش‌گر ممکن است یک حمله منع خدمت را روی یک وب‌گاه انجام دهد [11].

۲-۳- تدوین الزامات و سیاست‌های امنیتی نمایه‌ساز

نمایه‌سازی ناامن اطلاعات و داده‌ها، تهدیدی بزرگی برای

[1] Server

محرمانگی داده‌ها و اطلاعات یک وب‌گاه است. بنابراین داشتن راه‌کارهای لازم در زمینه نمایه‌سازی امن محتوا، برای هر وب‌گاهی الزامی است. از آنجا که جویش‌گرها نیز اطلاعات وب‌گاه‌ها را نمایه می‌کنند باید نمایه‌سازی خیلی امن‌تری داشته باشند. در بیشتر سرویس‌ها، نمایه‌سازی باید به‌صورت سلسله‌مراتبی انجام شود؛ یعنی ابتدا اطلاعات ورودی تجزیه و تحلیل‌شده، سپس با استفاده از الگوریتم‌های فشرده‌سازی و رمزنگاری، اطلاعات فشرده و رمز شوند و درنهایت در فهرست مربوطه ذخیره شدند. در ادامه سایر واحدهای استفاده‌شده برای نمایه‌سازی تشریح شده‌اند.

۱-۲-۳- ذخیره‌سازی داده‌ها در پایگاه داده

نمایه‌سازی باید به‌صورت سلسله‌مراتبی انجام شود؛ یعنی ابتدا اطلاعات ورودی تجزیه و تحلیل، سپس با استفاده از الگوریتم‌های فشرده‌سازی و رمزنگاری، اطلاعات فشرده و رمز و درنهایت در فهرست مربوطه ذخیره می‌شوند. در فرایند نمایه‌سازی اگر عملیات رمزنگاری اطلاعات مهم انجام نشود، مهاجم با استفاده از حملات مربوط به فهرست ناامن، فهرست مربوطه را کشف و به‌راحتی اطلاعات غیر مجاز را از فهرست کشف‌شده، بازیابی و مشاهده می‌کند.

ذخیره‌سازی امن داده‌ها در پایگاه داده مستلزم توجه به جنبه‌های مختلفی چون محرمانگی و صحت داده‌ها، کنترل دسترسی و نوع پایگاه داده است، که در ادامه به هر یک از این جنبه‌ها اشاره می‌کنیم.

محرمانگی: روش‌هایی برای حفاظت از محرمانگی و پنهان‌سازی داده‌های ذخیره‌شده در پایگاه داده‌ها وجود دارد که اغتشاش داده‌ها، رمزنگاری داده‌ها، تسهیم داده‌ها و بازیابی محرمانه اطلاعات، محدودکردن پرس‌وجوها از مهم‌ترین آنها هستند. در فرایند نمایه‌سازی داده‌ها، برای افزایش امنیت باید از گسسته‌سازی اجزای ذخیره‌سازی و پیاده‌سازی سازوکارهای کنترل دسترسی استفاده کرد تا نشت اطلاعاتی صورت نگیرد؛ زیرا با گسسته‌سازی اجزا اگر دسترسی غیر مجازی به داده‌ها صورت گیرد، فقط بخشی از داده‌ها آشکار خواهد شد (همچنین در گسسته‌سازی اجزا سیاست‌های امنیتی متفاوتی را روی هر جزء ذخیره‌سازی، می‌توان پیاده‌سازی کرد) [28]، [16]. شکل (۲) نحوه سطح‌بندی، گسسته‌سازی و واحدبندی یا پیمانه‌ای‌کردن منابع ذخیره‌سازی را نشان می‌دهد و هر کدام از منابع برای داده خاصی استفاده می‌شود.

(شکل-۲): امنیت در سطح سند و جمع‌آوری [28]

صحت داده: به‌طورمعمول در پایگاه داده‌های مبتنی بر SQL برای کنترل صحت داده‌ها بیشتر از محدودیت‌ها و رهانا[1] استفاده می‌شود. همچنین در پایگاه داده‌های بزرگ از الگوریتم‌های داده‌کاوی، برای استخراج دانش از پایگاه داده می‌توان استفاده کرد و بر اساس دانش حاصله میزان صحت داده‌های ذخیره شده را تعیین کرد.

کنترل دسترسی: کنترل دسترسی باید در سلسله سطوح مختلف پایگاه داده لحاظ شود تا کاربران و واحدهای سیستمی بر اساس وضعیت و اهمیت خود بتوانند به قسمت‌های مورد نیاز از پایگاه داده دسترسی داشته باشند. برای این منظور باید ابتدا کمینه امتیازهای اصلی که برای کاربران نیاز است، در سامانه تعریف و پیاده‌سازی شود؛ سپس در مراحل بعدی برای هر کاربر یا واحد بر اساس اعتبار و جایگاهی که در طول زمان کسب می‌کند، تفویض اختیار و امتیاز صورت گیرد. سیاست‌های کنترل دسترسی که برای پایگاه دادهٔ نمایه‌ساز جویش‌گرها تعریف می‌شود، بیشتر به واحدهای سیستمی داخل جویش‌گر برمی‌گردد. برای داشتن یک پایگاه داده امن باید موارد زیر به‌طور پیوسته کنترل تا تهدیدهای مربوطه در بالاترین سطح، خنثی و امنیت در سلسله‌مراتب پایگاه داده حفظ شود:

- کنترل دسترسی‌های قانونی برای جلوگیری از سوء استفاده؛
- اعمال سیاست‌های امنیتی برای جلوگیری از ترفیع حق دسترسی؛
- شناسایی آسیب‌پذیری‌های پلتفرم و وصله‌کردن آنها؛
- کنترل استنتاج‌های منطقی؛
- محدودکردن پارامترهای ورودی برای جلوگیری از تزریق کد
- پشتیبان‌گیری از اطلاعات؛
- غیر فعال‌کردن ویژگی‌های غیرضروری پایگاه داده‌ها

نوع پایگاه داده: به‌طورمعمول جویش‌گرها به‌دلیل

[1] Trigger

اینکه روزانه با داده‌های زیادی (جریانی از داده‌ها) مواجه می‌شوند نیاز دارند از پایگاه داده‌هایی استفاده کنند که عملیات ذخیره‌سازی داده‌ها در آنها سریع‌تر انجام شود و تکرار داده و همزمان‌سازی در آنها نیز به‌راحتی صورت گیرد. پایگاه داده‌های NoSQL برای این منظور مناسب هستند و برای ذخیره‌سازی داده‌های زیاد عملکرد خوبی دارند. در این پایگاه داده‌ها به‌دلیل اینکه لایه دسترسی به داده‌ها از لحاظ فیزیکی یا نرم‌افزاری کنترل می‌شود و تعداد سطوح دسترسی نسبت به پایگاه داده‌های SQL بیشتر است، سطح امنیتی بالاتری دارند و برای سرویس‌های جویش‌گرها که با داده‌ها و اطلاعات زیادی سروکار دارند، توصیه می‌شود [17].

به‌طورکلی سایر الزامات امنیتی که در پایگاه داده نمایه‌ساز و خزش‌گر می‌تواند استفاده شود، عباراتند از: ۱- محدودکردن کدهای پویای پایگاه‌داده‌ای. ۲- گسسته‌سازی و دانه‌بندی‌کردن منابع ذخیره‌سازی برای کاهش نشت اطلاعات در صورت نفوذ به سیستم. ۳- رمزگذاری فایل‌های داده برای برقراری محرمانگی داده‌ها. ۴- رمزگذاری و احراز هویت خدمتگزار و مشتری. ۵- رمزگذاری و احراز هویت بین خوشه‌ای. ۶- بهره‌گیری از روش‌های جلوگیری از تزریق کد مثل WAVES, DBC-Checker, WebSSARI, Java Static SQL DOM, Tainting, SecuriFly [21]، [22]. ۷- دقت در ذخیره‌سازی داده‌های موجود در هر عنصر پایگاه داده (صحت داده). ۸- ردگیری عامل انجام عملیات در پایگاه داده (نظارت‌پذیری[1]). ۹- کنترل‌کردن اسکریپت‌ها و ورودی‌های اجرایی مخرب و مقایسه آنها با فهرست‌های کنترلی از پیش تعیین‌شده. ۱۰- ارائه ساختارها و واحدهای امنیتی مثل دیوار آتش به‌صورت سلسله‌مراتبی برای جلوگیری از ارتباط مستقیم به پایگاه داده‌ها. ۱۱- اعتبارسنجی اندازه، قالب، بازه و نوع داده‌های ورودی. ۱۲- مدیریت قوی کلیدهای پایگاه داده‌ای و بهره‌گیری از سامانه‌های مدیریت کلیدی که از واسط پایگاه داده و نوع کلیدهای آن، پشتیبانی کند. امروزه از واحدهای سخت‌افزاری امن مثل HSM[2] برای این منظور استفاده می‌شود. ۱۳- اعمال روش‌های کنترل حملات تزریق کد: پرس‌وجوهای پارامتربندی‌شده (Parameter Query)، پردازه‌های ذخیره‌شده (Store Procedure)، استفاده از دستور Escaping، استفاده از فهرست سفید (White List)، اعطای کمینه مجوز (Least Privilege)، صدور مجوز برای استفاده از دستورهای (Grant Command). ۱۴- پشتیان‌گیری از داده‌ها و اطلاعات ذخیره‌شده برای افزایش دسترس‌پذیری. ۱۵- استفاده از ساختارهای امنیتی مثل هانی پات‌ها[3] برای سابقه‌گیری[4] و به‌دست‌آوردن اطلاعات بیشتر از مهاجمان و ارائه سناریوهای از پیش تعیین‌شده به آنها. ۱۶- کنترل آسیب‌پذیری‌های پلتفرم و غیرفعال‌کردن خدمات اضافی. ۱۷- کنترل ورودی‌های نامعتبر توسط توابع امنیتی کتابخانه‌ای، برای جلوگیری از حملات تزریق کد. ۱۸- استفاده از پروتکل‌های امنیتی (SSL, TLS, IPSec) برای برقراری ارتباط امن بین خدمت‌گزارهای خزش‌گر و پایگاه داده. ۱۹- محدودساختن هر واحد یا کاربر به داده هایی که مجاز به دسترسی یا تغییر آنها است (کنترل دسترسی). ۲۰- ایمنی ساختار پایگاه داده در مقابل انتشار تغییرات ناخواسته (جامعیت منطقی پایگاه). ۲۱- ایمنی داده های موجود در پایگاه داده نسبت به مخاطرات فیزیکی (جامعیت فیزیکی پایگاه). ۲۲- دسترسی به پایگاه داده در همه شرایط (دسترس‌پذیری). ۲۳- بررسی هویت کاربران در ارتباط با ردگیری نظارتی و اجازه دسترسی به داده‌های خاص (تصدیق اصالت و هویت شناسی کاربران). ۲۴- توجه به مسائل سازگاری اطلاعات ذخیره‌شده در پایگاه داده (استفاده از محیط‌های ابری و مجازی‌سازی باعث افزایش ناسازگاری پایگاه داده می‌شود که این مسائل باید توسط تیم امنیت مدیریت شود).

۲-۲-۳- موازی‌سازی فرآیندها

جویش‌گرها به‌دلیل اینکه از معماری توزیع‌شده بهره می‌برند، بیشتر با این چالش مواجه هستند؛ زیرا اگر موازی‌سازی فرایندها به‌درستی کنترل نشود، زمانی که کاربر یک جستجو انجام می‌دهد، انتظار دارد در کمترین زمان ممکن پاسخ مناسب را دریافت کند و این زمان با موازی‌سازی فرایندها در جستجوهای پایگاه داده جویشگر در بهینه‌ترین حالت قابل دستیابی است. بنابراین باید به مسائل موازی‌سازی و همزمان‌سازی توجه ویژه‌ای داشت. به‌طورمعمول با استفاده از فریمورک‌هایی مثل هدوپ[5] مسائل مربوط به مدیریت محتوا

[1] Auditability
[2] Hardware Security Module (HSM)
[3] Honeypot
[4] Log
[5] Hadoop

و پردازش‌های موازی روی معماری‌های توزیع‌شده به‌خوبی انجام می‌شود و در بیشتر زمان‌ها چالش‌های موازی‌سازی قابل کنترل می‌باشند.

۳-۲-۳- فاکتورهای طراحی نمایه‌ساز

پیاده‌سازی الزامات زیر در هنگام طراحی نمایه‌ساز جویشگرها باعث امن‌شدن معماری آن خواهد شد:

- قابلیت اطمینان نرم‌افزاری؛
- افزونگی محتوا؛
- توزیع‌شدگی واحدهای نمایه‌سازی برای افزایش دسترس‌پذیری؛
- رفع خطاهای مربوط به نمایه‌سازی در انزوا؛
- افزایش تحمل‌پذیری خطا؛
- خدمت‌گزارهای مناسب جهت کاهش هزینه‌های محاسباتی؛
- سامانه‌های پشتیبان؛
- مراکز داده مناسب

۴-۲-۳- واحد جستجو

دلایل اصلی که باعث بروز آسیب‌پذیری در سکوهای واحدهای جستجو می‌شود، ضعف‌های نرم‌افزاری است که در هنگام توسعه به‌وجود آمده‌اند. توصیه‌ای که برای این منظور می‌توان بیان کرد، اطمینان از به‌روزبودن بسته‌های نرم‌افزاری و آزمایش پیوسته آنها است تا درصورت وجود آسیب‌پذیری، سریعا وصله امنیتی آن نصب شود. همچنین فیلترکردن تمامی ورودی کاربران (به‌خصوص اجازه ندهید که متا نویسه‌ها، داخل ورودی‌های کاربران قابل درج باشد). برای مقابله با کدهای اجرایی در واحد جستجو، داده‌های غیر قابل اعتماد از دستورها و پرس‌وجوها را می‌توان جدا کرد. به این نکته نیز باید توجه کرد که سیاست‌های دفاعی اپلیکیشن‌های تحت وب، قابل استفاده در واحد جستجو است.

۵-۲-۳- ساختمان داده نمایه‌ساز

یکی از مسائل مهمی که در نمایه‌سازی جویش‌گر اهمیت دارد، نحوه انتخاب ساختمان دادهٔ نمایه‌ساز است. انواع مختلفی از ساختارها و ساختمان داده‌ها وجود دارد که برای نمایه‌سازی می‌توان استفاده کرد. مناسب‌ترین ساختمان داده‌ای که برای نمایه‌سازی اطلاعات در جویش‌گرها می‌توان استفاده کرد، نمایه معکوس است.

۶-۲-۳- دورهٔ به‌روزرسانی نمایه‌ساز

رسیدن به یک مصالحه برای زمان دوره به‌روزرسانی نمایه‌ساز به‌طوری‌که تازگی اطلاعات همواره در حد مطلوب باشد، برای هر جویش‌گری یک الزام اساسی است. همچنین دوره به‌روزرسانی باید به‌گونه‌ای تنظیم شود که صفحات وب‌گاه‌های آلوده نیز در کمترین زمان ممکن کشف شوند. در بیشتر جویش‌گرهای بزرگ دوره‌های به‌روزرسانی نمایه‌ساز ثانیه‌ای یا ساعتی است.

۷-۲-۳- سئوی منفی

در جویش‌گرهای بزرگی مثل گوگل و بینگ سیاست‌هایی تعریف شده است که اگر وب‌گاه‌ها مسائل مربوط به سئوی کلاه‌سیاه را در وب‌گاه‌ها لحاظ کنند، آنگاه رتبه آنها کاهش یافته و یا ممکن است از فهرست نمایه‌سازی حذف شوند. بنابراین اگر جویشگری در برابر سئوی منفی ایمن نباشد، آنگاه احتمال حذف وب‌گاه‌های درست از فهرست نمایه‌سازی وجود دارد. به‌همین دلیل جویش‌گرها باید سیاست‌های دفاعی خود را همیشه به‌صورت دستی یا خودکار در فرایند جستجو پیاده‌سازی کنند تا هیچ وب‌گاه درست و سالمی با مشکل مواجهه نشود.

۸-۲-۳- سیاست‌های مقابله با صفحات و وب‌گاه‌های آلوده

به‌طورکلی در فرآیند نمایه‌سازی ممکن است، صفحاتی نمایه شوند که به‌هیچ‌وجه ارزش محتوایی برای کاربران نداشته باشند و رجوع به این صفحات نارضایتی کاربران را به‌همراه دارد. مؤلفه نمایه‌ساز درصورتی‌که قابلیت کشف صفحات آلوده و مشکوک را داشته باشد، در زمان نمایه‌سازی با همکاری مؤلفه رتبه‌بند از دو رویکرد زیر می‌تواند استفاده کند.

- اگر جویش‌گر، یک صفحه وب را آلوده تشخیص داد، آن صفحه را می‌تواند نمایه‌سازی نکند و به‌طورکامل آن را کنار بگذارد. با این کار کاربران به‌هیچ‌وجه از طریق جویش‌گر به صفحه آلوده هدایت نمی‌شوند؛ اما مشکلی که در این رویکرد وجود دارد، مسئله تشخیص اشتباه است؛ زیرا اگر یک صفحه سالم به‌اشتباه، آلوده تشخیص

داده شود، آنگاه احتمال نمایه‌سازی آن صفحهٔ سالم به‌طورتقریبی صفر است. بنابراین برای جلوگیری از این اتفاق رویکرد دوم پیشنهاد می‌شود.

- اگر جویش‌گر، صفحه وب را آلوده تشخیص داد می‌تواند با همکاری مؤلفه رتبه‌بند، آن صفحه را با میزان رتبه کمتری نمایه‌سازی کند؛ زیرا با این عمل کاربران با احتمال کمتری صفحات مشکوک و آلوده را مشاهده می‌کنند و اگر کاربران به پیوند صفحات مشکوک و آلوده رسیدند، جویش‌گر باید با صدور یک پیام هشدار، هدف چنین صفحاتی را برای کاربران آشکار کند.

۳-۳- الزامات و سیاست‌های امنیتی مؤلفه رتبه‌بند

درصورتی‌که مؤلفه رتبه‌بند یک جویش‌گر ضعیف و آسیب‌پذیر باشد، تهدیدی بزرگی برای وب‌گاه‌های سالم و درستکار به‌شمار می‌رود؛ زیرا چنین وب‌گاه‌هایی در زیر صفحات هرزنامه و مخرب دفن می‌شوند و کاربران به‌سختی قادر به مشاهده آنها هستند؛ بنابراین در این بخش سیاست‌ها و الزامات امنیتی کلی که باید در مؤلفه رتبه‌بند پیاده‌سازی شود، ارائه شده است.

۱-۳-۳- سیاست‌های مربوط به صفحات هرزنامه

هر صفحهٔ وبی که از طریق روش‌های غیر مجاز مثل ریسندگی محتوا، مزارع پیوند، چاشنی کلمه کلیدی و غیره افزایش رتبه پیدا کند و در نتایج جستجو ظاهر شود، به‌عنوان صفحه هرزنامه شناخته می‌شود. مبارزه با صفحات هرزنامه در جویش‌گرها باید جزء اولویت‌های اصلی باشد؛ زیرا آنچه که باعث محبوبیت یک جویش‌گر می‌شود ارائه نتایج مرتبط و فریب‌نخوردن کاربران است. بنابراین می‌توان از الگوریتم‌های اکتشافی[1] و ژنتیک[2] که سرعت بالایی در پردازش دارند در تجزیه و تحلیل کدهای تحت وب برای کشف صفحات هرزنامه استفاده کرد. به‌طورمعمول مقابله با هرزنامه‌ها در سطح رتبه‌بند صورت می‌گیرد؛ ولی منطقی‌ترین حالت پیاده‌سازی سیاست‌های امنیتی آن در سطوح مختلف (خزش‌گر، نمایه‌ساز و رتبه‌بند) جویش‌گر است به‌طوری‌که بخش نمایه‌ساز و رتبه‌بند مناسب‌ترین سطوح برای مقابله با هرزنامه‌ها است. درواقع این دو سطح با همکاری یکدیگر به بهترین نحو ممکن با هرزنامه‌ها می‌توانند مقابله کنند.

۲-۳-۳- سیاست‌های مقابله با ریسندگی محتوا و ربودن پراکسی

رویکردی که برای این منظور می‌توان استفاده کرد بدین صورت است، اگر یک صفحه در زمان T ایجاد شده و همان صفحه در زمان T+n بازنویسی شده است، جویش‌گر باید زمان ایجاد صفحات را در نظر بگیرد تا پی‌ببرد که مسئله تکراری بودن و بازنویسی محتوا مربوط به کدام صفحه است. اگر زمان ایجاد صفحات به‌عنوان یک پارامتر در رتبه‌بندی صفحات لحاظ شود، آنگاه مسئله تکراری‌بودن صفحات وب و محتواها که به وسیله ریسندگی محتوا یا ربودن پراکسی ایجاد می‌شود، قابل حل خواهد بود. همچنین برای جلوگیری از هک پراکسی جویش‌گرها باید اتصالاتی را که از خدمت گزارهای پراکسی به وب‌گاه‌ها وارد می‌شود، در فرایند رتبه‌بندی صفحات لحاظ کنند و از تأثیر زیاد آنها در رتبه‌بندی جلوگیری شود؛ زیرا اگر پیوندهایی که از خدمت گزارهای پراکسی به وب‌گاه‌ها داده می‌شوند، به‌عنوان پیوند معمولی در نظر گرفته شود، آنگاه یک وب‌گاه یا صفحه هرزنامه با ربودن پراکسی به بالاترین رتبه در نتایج جویش‌گرها می‌رسد.

۳-۳-۳- سیاست‌های مقابله با صفحات پنهان

جویش‌گرها با سیاست و رویکرد تغییر نام یا تغییر نشانی IP اسپایدرها، می‌توانند وب‌گاه‌هایی را که محتوای متفاوتی ارائه می‌کنند، تشخیص دهند؛ سیاست تغییر نشانی با خزش‌گرهای توزیع‌شده که در نقاط جغرافیایی متفاوتی قرار گرفته‌اند محقق می‌شود؛ زیرا با این کار وب‌گاه‌ها جویش‌گر را به‌عنوان یک کاربر معمولی شناسایی می‌کنند. در این لحظه جویش‌گر چون از قبل محتوای متفاوتی را از همان وب‌گاه مشاهده کرده به‌راحتی پی می‌برد که صفحه مورد نظر یک صفحه پنهان است.

۴-۳-۳- سیاست‌های مقابله با مزرعه پیوند

- محدودکردن چگالی پیوندها برای رتبه‌بندی صفحات وب
- درنظرگرفتن مبدأ و مقصد پیوندها و شناسایی همسایه‌های بد

[1] Heuristic
[2] Genetic

• جریمه‌نکردن وب‌گاه‌های معتبر به‌دلیل پیونددادن از وب‌گاه‌های هرزنامه؛ زیرا مهاجم ممکن است به‌دلایلی از ربودن پراکسی استفاده و از وب‌گاه جعلی موجود در پراکسی که محتوای نامطلوب دارد، پیوندی به وب‌گاه اصلی و معتبر ایجاد کند. بنابراین اگر جویشگر در برابر چنین تهدیدی، سیاست و ساختار دفاعی لازم را نداشته باشد، آنگاه رتبه وب‌گاه اصلی به‌دلیل داشتن پیوند ورودی از همسایه بد تنزل خواهد کرد.

۵-۳-۳- سیاست‌های مقابله با چاشنی کلمه کلیدی

• اعمال چگالی ۷ الی ۱۰ درصدی از کلمات کلیدی برای محاسبه رتبه هر صفحه وب [14]

• کشف کلمات کلیدی مخفی در کدهای وب و نادیده‌گرفتن آنها در رتبه‌بندی صفحات وب (در الگوریتم رتبه‌بندی جریمه‌هایی برای چنین کارهایی می‌توان اعمال کرد)

۶-۳-۳- سیاست‌های مقابله با صفحات درگاه

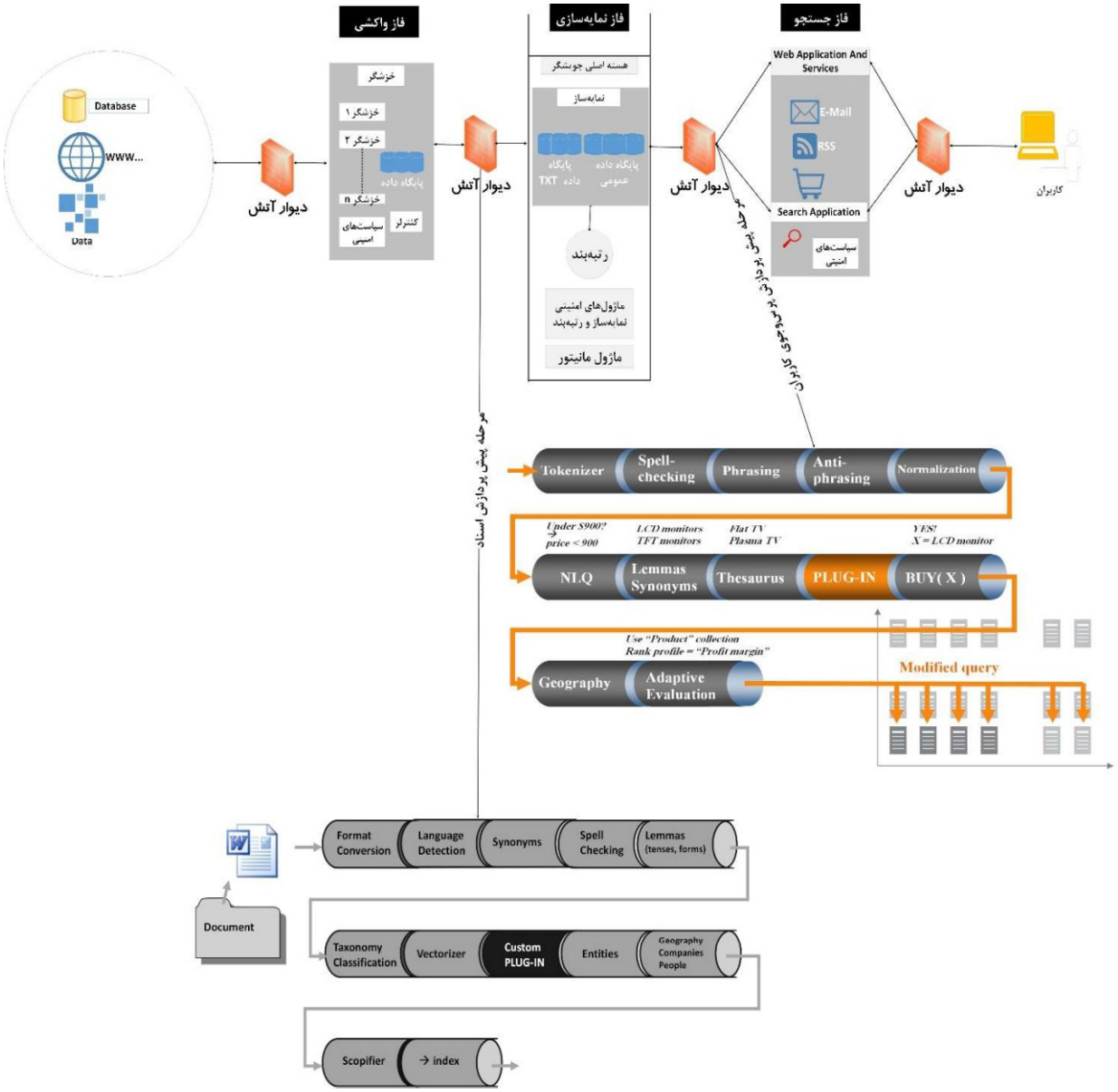
جویش‌گرها با تحلیل و پالایش‌کردن صفحات وب میزان چگالی بهینگی عبارات را می‌توانند محاسبه کنند و صفحاتی که روی یک عبارت و کلمه خاصی چگال‌تر هستند به‌عنوان صفحه درگاه شناخته شوند؛ بنابراین مهم‌ترین پارامتری که در شناخت چنین صفحاتی تأثیر دارد، فراوانی عبارات و کلمات خاص است. همچنین برخی از صفحات درگاه ممکن است، دارای اسکریپت‌های باشند که به‌محض اینکه کاربر به چنین صفحاتی رجوع می‌کند، آن اسکریپت اجرا شود. درواقع مبارزه با این صفحات باید با رویکردهای متفاوتی صورت گیرد که تمام حالت‌های مخرب توسط جویش‌گر شناسایی شود (هر سیاست امنیتی که برای صفحات هرزنامه استفاده می‌شود، قابل پیاده‌سازی برای این صفحات نیز است).

۷-۳-۳- سیاست‌های مقابله با صفحات و وب‌گاه‌های آلوده

اگر چه مراحل مناسب برای کشف و مقابله با صفحات آلوده در زمان فعالیت خزش‌گر و نمایه‌ساز است؛ اما پیاده‌سازی سامانه دفاعی در مرحله رتبه‌بندی نیز یک امتیاز می‌تواند محسوب شود؛ زیرا با این کار اگر یک وب‌گاه سالم به دلایل احتمالی و False Positive الگوریتم‌ها و سامانه‌های دفاعی در سطوح خزش‌گر و نمایه‌ساز، به‌اشتباه آلوده تشخیص داده شود، آنگاه شانس نمایه‌سازی آن به‌طورتقریبی صفر می‌شود به‌همین دلیل وجود سامانه دفاعی برای مقابله با صفحات آلوده در سطح رتبه‌بند، نیز پیشنهاد می‌شود. البته پیشنهاد نهایی یک حالت ترکیبی است که در قسمت سیاست‌های امنیتی نمایه‌ساز ارائه شد و با آن رویکرد، مقابله با صفحات آلوده بهترین عملکرد را خواهد داشت. به‌یقین مسئله رتبه‌بندی صفحات وب چیزی نیست که با درنظرگرفتن چند تهدید و اعمال راه‌کارهای امنیتی برای آنها بهترین رتبه‌بندی را بتوان داشت. زیرا مسئله رتبه‌بندی و الگوریتم‌های آن باید متناسب با زمان تغییر کند تا در برابر انواع تهدیدهای جدید ایمن شود و درنهایت مناسب‌ترین صفحات در نتایج جستجو نشان داده شود [20]، [19]، [18].

۴- امنیت در معماری جویش‌گرهای بومی

در بخش‌های قبل تهدیدها و آسیب‌پذیری‌های مؤلفه‌های اصلی جویش‌گر مشخص و در همان راستا سیاست‌ها و الزامات امنیتی تعیین شدند؛ به‌طوری که با امن‌شدن مؤلفه‌ها، معماری کلی جویشگر نیز امن خواهد شد. به‌همین دلیل در این بخش امنیت معماری جویش‌گرها بر اساس مؤلفه‌ها تشریح شده است. امنیت در معماری جویش‌گرها باید در سه مرحله در نظر گرفته شود که این سه مرحله شامل مرحلهٔ واکشی، مرحلهٔ نمایه‌سازی و رتبه‌بندی در محیط خدمت‌گزارهای داخلی جویش‌گر و مرحلهٔ جستجو است. شکل (۳) معماری کلی و عمومی جویشگر را نشان می‌دهد. همان‌طور که در شکل (۳) مشاهده می‌شود، وقتی کاربران از طریق مرحلهٔ جستجو، پرس‌وجوی خود را ارسال می‌کنند، قبل از اینکه به مرحلهٔ نمایه‌سازی برسد، باید توسط "مرحله پیش پردازش پرس‌وجوی کاربران" وارسی و سپس به سطح نمایه‌سازی ارسال تا مناسب‌ترین پاسخ به کاربران ارائه شود. همچنین در مرحلهٔ واکشی اسناد نیز چنین ساختاری وجود دارد که قبل از اینکه اسناد وب‌گاه‌ها به مرحلهٔ نمایه‌سازی برسد، با "مرحله پیش‌پردازش اسناد" تجزیه و تحلیل می‌شود تا اصطلاحات و اطلاعات اشتباه نمایه‌سازی نشوند؛ زیرا اگر چنین اطلاعات اشتباهی به مرحله نمایه‌سازی برسد، تطبیق دادن پرس‌وجوهای کاربران با آنها سخت‌تر می‌شود [23]، [18].

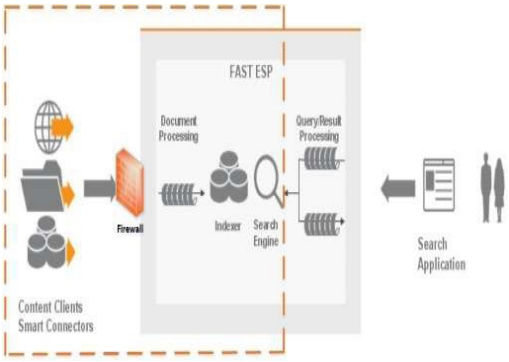


(شکل-۳): مراحل عمومی جویش‌گرهای بومی

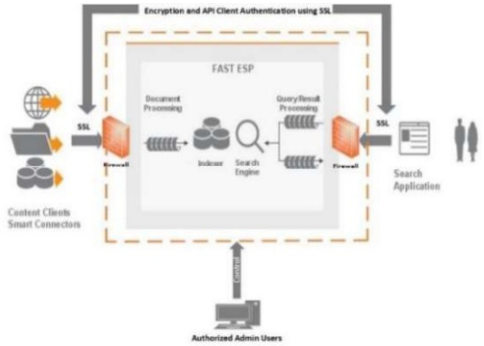
شکل‌های (۴) تا (۶) امنیت سلسله‌مراتبی را در معماری هسته جویشگر بهتر نشان می‌دهد. هر کدام از مرحلهٔ واکشی، نمایه‌سازی و جستجو در این شکل‌ها با خط‌چین مشخص شده‌اند. مهم‌ترین مسئله امنیتی که در سه شکل مورد نظر، یک تهدید بزرگ برای جویش‌گر به‌شمار می‌رود، دسترسی به پایگاه داده‌های نمایه‌ساز است [۲۴]،[۲۳].

اگر ارتباطات از سمت مرحلهٔ واکشی و مرحلهٔ جستجو کنترل و مدیریت نشود، مهاجم از طریق این دو درگاه به پایگاه داده نمایه‌ساز دسترسی می‌تواند داشته باشد. مهم‌ترین واحد امنیتی مناسب برای این درگاه‌ها استفاده از دیوار آتش‌های معمولی یا هوشمند است؛ زیرا بیشتر محصولات امنیتی مثل دیوار آتش‌های لایه سه، چهار و هفت شبکه، سیاست‌های امنیتی کنترل پورت، تجزیه درخواست‌ها، شناسایی داده‌های مخرب، کنترل حملات منع خدمت، کنترل ترافیک‌های عبوری و غیره را انجام می‌دهند.

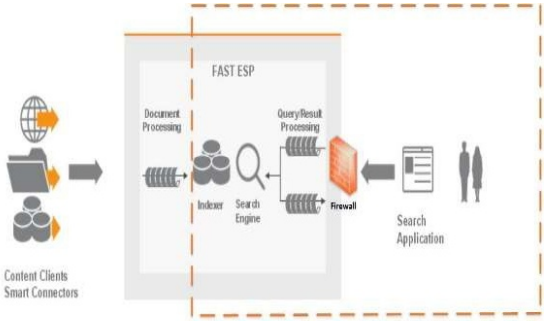
شکل (۴) امنیت در مرحلهٔ واکشی را مشخص می‌کند؛ زیرا در این مرحله باید مسائل مربوط به سیاست‌های خزش، مبارزه با سئوی کلاه‌سیاه و وب‌گاه آلوده، امنیت پایگاه داده‌ها، فاکتورهای طراحی در نظر گرفته شود؛ زیرا پیاده‌سازی ساختارهای امنیتی در این مرحله باعث می‌شود که اثرات خراب‌کارانه آن به مراحل و سطوح دیگر



(شکل-۴): امنیت در مرحلهٔ واکشی [۲۴]



(شکل-۵): امنیت در مرحلهٔ نمایه‌سازی خدمت‌گزارهای داخلی [۲۴]



(شکل-۶): امنیت در مرحلهٔ اپلیکیشن جستجو [۲۴]

وارد نشود؛ زیرا با بهره‌گیری از ساختارهای امنیتی در سطوح و مراحل مختلف خطرات احتمالی آنها در همان سطوح خنثی می‌شود. در مرحلهٔ واکشی علاوه‌بر دیوار آتش انواع سیاست‌های مقابله با سئوی منفی، صفحات و وب سایت‌های آلوده، سیاست‌های خزش برای کنترل تازگی اطلاعات و حذف صفحات آلوده، وارسی پیوندهای عبوری و غیره را می‌توان انجام داد.

شکل (۵) وضعیت ساختارهای امنیتی را در مرحله یا سطح نمایه‌سازی و رتبه‌بندی نشان می‌دهد؛ زیرا از آنجا که مرحلهٔ نمایه‌سازی مهم‌ترین قسمت و هسته اصلی یک جویش‌گر محسوب می‌شود، توجه به سیاست‌ها و ساختارهای امنیتی آن همواره اهمیت دارد. مسائل امنیتی که باید در این قسمت به آن توجه داشت، شامل امنیت پایگاه داده، امنیت واحد جستجو، امنیت واحد تجزیه‌کننده، فاکتورهای طراحی و سیاست‌های مقابله با وب‌گاه‌های آلوده و سئوی منفی است. علاوه‌بر این مرحلهٔ نمایه‌سازی باید توسط واحدهایی مثل دیوار آتش از سمت مرحلهٔ واکشی و مرحلهٔ جستجو حفاظت تا میزان دسترسی‌های عمومی محدود شود. مهم‌ترین مسئله امنیتی که باید در مرحلهٔ نمایه‌سازی در نظر گرفت، امن‌کردن پایگاه داده نمایه‌ساز است؛ زیرا با دسترسی به پایگاه داده نمایه‌ساز امنیت جویش‌گر به خطر می‌افتد. بنابراین یکی از واحدهای اصلی امنیتی که باید برای این امر در نظر گرفت، استفاده از دیوار آتش است. شکل (۶) امنیت جویش‌گر را در مرحلهٔ اپلیکیشن جستجو نشان می‌دهد و کاربران نخستین سطحی را که در جویش‌گر می‌بینند، واسط کاربری جستجو[1] است. مسائل امنیتی که در مرحلهٔ جستجو باید به آن توجه شود، شامل سیاست‌های مقابله با حملات تحت وب (مثل SQLI, XSS, Buffer OverFlow, DOS)، گواهی‌نامه‌های امنیتی و غیره است (شایان ذکر است که در این مقاله بررسی امنیت در مرحلهٔ جستجو یا اپلیکیشن تحت وب مد نظر نبوده و بیشترین تمرکز روی سه مؤلفه اصلی یعنی خزش‌گر، نمایه‌ساز و رتبه‌بند بوده است).

جدول (۱) انواع زیرواحدهای مؤلفه‌های اصلی جویش‌گرها را از ابعاد مختلف امنیتی بررسی می‌کند، در این جدول سعی شده است به‌گونه‌ای تکمیل شود که مؤلفه‌های هر جویش‌گری از جمله پارسی‌جو و یوز را شامل شود. این اطلاعات بر اساس استاندارد ITU-T X.805 که نگاهی کلی به ابعاد امنیتی اصلی دارد تکمیل شده است. طبق این جدول تمام زیرواحدها باید جنبه‌های امنیتی دسترس‌پذیری، کنترل دسترسی، تصدیق اصالت و امنیت ارتباط را فراهم کنند. دسترس‌پذیری باید در تمام چرخه حیات جویش‌گر همواره برقرار باشد تا جویش‌گر در هر شرایط و زمانی پاسخ‌گوی کاربران باشد.

محرمانگی داده‌ها باید برای پایگاه داده خزش‌گر و نمایه‌ساز همواره برقرار باشد؛ زیرا اگر نشت اطلاعاتی صورت گیرد آنگاه مهاجمان نمی‌توانند به داده متن آشکار دسترسی داشته باشند، بنابراین وجود سازوکارهای رمزنگاری برای

[1] Search Interface

برقراری محرمانگی همواره الزامی است. اگر چه خود الگوریتم‌های خزش‌گر، نمایه‌ساز و رتبه‌بند باید محرمانه باشند، اما ممکن است که یک جویش‌گر از الگوریتم‌های متن باز استفاده کند و درعمل محرمانگی برای آنها معنا نداشته باشد؛ ولی اگر الگوریتم‌ها بومی‌سازی شده باشد، آنگاه محرمانگی برای آنها حائز اهمیت خواهد بود.

صحت داده‌های ذخیره‌شده در پایگاه داده همواره مهم است؛ زیرا اگر اطلاعات متناقضی در پایگاه داده وجود داشته باشد، آنگاه پاسخ به پرس‌وجوهای کاربران بر اساس اطلاعات اشتباه خواهد بود؛ به‌طوری‌که اگر جویش‌گر از پایگاه داده‌های توزیع‌شده استفاده کند، وجود صحت داده اهمیت بیشتری برای پایگاه داده خواهد داشت.

مسئله حریم خصوصی بیشتر در خدمات تحت وب اهمیت دارد و مؤلفه‌های اصلی جویش‌گر ارتباط کمتری با آن دارند؛ اما زیرواحد جستجو و الگوریتم خزش باید به جنبه حریم خصوصی نیز توجه کنند؛ زیرا واحد جستجو اگر با اطلاعات دریافتی از کاربران به پرس‌وجوها پاسخ دهد، آنگاه باید مرحله گمنامی کاربران را در حین جستجو در نظر بگیرد تا ردپای از کاربران روی وب سایت‌ها ثبت نشود. همچنین الگوریتم‌های خزش باید حریم خصوصی وب‌گاه‌ها را نیز رعایت کنند تا اطلاعات خصوصی که مالکان وب‌گاه مایل به نشان‌دادن آنها نیستند، افشا نشود (این مسئله بیشتر به سیاست‌های امنیتی الگوریتم‌های خزش جویش‌گر بستگی دارد).

جنبه امنیتی انکارناپذیری باید توسط الگوریتم‌های خزش حفظ شود؛ زیرا ممکن است که یک خزش‌گر به وب‌گاهی رجوع کند و به‌دلیل یک آسیب‌پذیری در خزش‌گر، حمله تزیق کد را روی وب‌گاه مربوطه انجام دهد. در این زمان مالک وب‌گاه بر اساس ردپای جویش‌گر باید قادر به شناسایی آن باشد تا جویش‌گر نتواند عمل انجام‌شده را انکار کند.

(جدول-۱): بررسی مؤلفه‌های جویش‌گرهای بومی از ابعاد امنیتی

ابعاد امنیتی	خزش‌گر					نمایه‌ساز						رتبه‌بند			
	پایگاه داده	الگوریتم مقابله با وب سایت‌های آلوده	الگوریتم مقابله با سئو منفی کلاه سیاه	الگوریتم خزش	خدمتگزارهای خزش‌گر	پایگاه داده‌ها	الگوریتم مقابله با سئو منفی کلاه سیاه	الگوریتم مقابله با وب سایت‌های آلوده	الگوریتم نمایه سازی	واحد جستجو	واحد تجزیه کننده	خدمتگزارهای نمایه ساز	الگوریتم مقابله با وب سایت‌های آلوده	الگوریتم رتبه‌بندی	الگوریتم مقابله با سئو منفی کلاه سیاه	خدمتگزارهای رتبه‌بندی
دسترس پذیری	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
محرمانگی	√	×	×	×	√	√	×	×	×	×	×	√	×	×	×	√
حریم خصوصی	×	×	×	√	×	×	×	×	×	√	×	×	×	×	×	×
کنترل دسترسی	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
صحت داده‌ها	√	×	×	×	√	√	×	×	×	×	×	√	×	×	×	√
تصدیق اصالت	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
امنیت ارتباط	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
انکار ناپذیری	×	×	×	√	×	×	×	×	×	×	×	×	×	×	×	×

۵- نتیجه‌گیری

در این مقاله انواع حملات و آسیب‌پذیری‌هایی که هر کدام از مؤلفه‌های خزش‌گر، نمایه‌ساز و رتبه‌بند، یک جویشگر را تهدید می‌کنند، مورد بررسی قرار گرفت؛ سپس متناسب با آسیب‌پذیری‌های شناسایی‌شده، الزامات و سیاست‌های امنیتی برای هر کدام از مؤلفه‌ها به‌طور مجزا ارائه شد و نشان داده شد که با امن‌شدن این سه مؤلفه، معماری کلی جویش‌گر امن خواهد شد. بخش پایانی مقاله به معماری امن جویش‌گر اشاره شد، در معماری پیشنهادی امنیت معماری

جویش‌گر در سه مرحلهٔ اصلی به نام مرحلهٔ جستجو، مرحلهٔ نمایه‌سازی و رتبه‌بندی و مرحلهٔ واکشی بررسی و ارتباط ابعاد مختلف امنیتی با زیر واحدهای مؤلفه‌های اصلی جویش‌گر، ارائه شد. همچنین در ادامه پیشنهاداتی برای جویش‌گرهای بومی ارائه شده است که پیاده‌سازی آنها منجر به بهبود عملکرد، افزایش محبوبیت، نتایج بهتر و ارتقاء امنیت (وب سایت‌ها و کاربران) را برای جویش‌گرهای بومی به همراه دارد.

۱- پیاده‌سازی ابزارهای مدیریت وب‌گاه‌ها مثل Google Webmaster Tools در جویش‌گرهای بومی؛ زیرا چنین ابزاری به مالکان وب‌گاه این اختیار را می‌دهد که وب‌گاه خود را متناسب با الگوریتم‌های رتبه‌بندی جویش‌گرهای بومی بهینه کنند و این ابزار برای مالکان وب‌گاه‌های تجاری بسیار جذاب خواهد بود؛ زیرا جویش‌گرهای بزرگی مثل گوگل سیاست‌های کلی پاداش و سیاست‌های جریمه الگوریتم‌های رتبه‌بندی خود را ارائه می‌کنند تا مالکان وب‌گاه‌ها راحت‌تر در راستای آن سیاست‌ها بتوانند اقدام کنند.

۲- بهره‌گیری از پارامترهای رتبه‌بندی بیشتر برای رتبه‌دهی صفحات و وب‌گاه‌ها. زیرا طبق سند [۹] تعداد پارامترهایی که در جویش‌گرهای بومی استفاده می‌شود، نسبت به جویش‌گرهای جهانی کمتر است.

۳- ارائه ساختارها و سیاست‌های دفاعی مناسب برای مقابله با سئوی کلاه‌سیاه. زیرا طبق بررسی‌های صورت‌گرفته و بر اساس سند [۹]، سیاست مطلوبی که از سئوی کلاه‌سیاه جلوگیری کند، در ساختار مؤلفه‌های اصلی (مثل خزش‌گر، نمایه‌ساز و رتبه‌بند) جویش‌گرهای بومی وجود ندارد.

۴- جویش‌گرهای بومی در مواجه با صفحات سمی[1] و آلوده، در مقایسه با جویش‌گرهای جهانی عملکرد و سیاست مناسبی ندارند و سیاست و ساختار امنیتی برای پالایش و تحلیل کدهای مخرب وب در آنها دیده نمی‌شود. همچنین ارائه هشدارهای امنیتی به کاربران در هنگام ملاقات صفحات آلوده یکی از الزامات اساسی برای جویش‌گرهای بومی به‌شمار می‌رود که باید اقدامات لازم برای پیاده‌سازی این مورد صورت گیرد.

۵- جویش‌گرهای بومی مانند جویش‌گرهای بزرگی مثل گوگل باید قابلیتی داشته باشند که کاربران بتوانند صفحات وب هرزنامه را به آنها گزارش دهند و با بررسی و تحلیل دقیق‌تر آن صفحات، تصمیم نهایی را برای حذف چنین صفحاتی از نتایج جستجو اتخاذ کنند. همچنین جویش‌گرهای بومی باید گزینه‌ای داشته باشند که به‌صورت برخط[2] میزان رتبه و درصد هرزنامه وب‌گاه‌ها را نشان دهند.

۶- جویش‌گرهای بومی باید از زیرساخت‌های امن و قابل اطمینان برای سه واحد خزش‌گر، نمایه‌ساز و رتبه‌بند استفاده کنند تا بستر لازم برای جریان‌های داده‌ای برخط زیاد فراهم شود.

۷- جویش‌گرهای بومی در استراتژی‌های انتخاب محتوا ضعیف هستند؛ بنابراین برای هر محتوایی و پاسخ به هر تقاضایی باید از استراتژی و خزش‌گر مربوط به همان محتوا استفاده شود. به‌عنوان مثال برای محتواهای علمی، عمومی، خاص و غیره رویکرد مجزایی باید وجود داشته باشند.

به‌طورکلی به‌دلیل این‌که جویش‌گرها، از جمله جویش‌گرهای بومی از مؤلفه‌های زیادی برای ارائه سرویس استفاده می‌کنند و هر کدام از مؤلفه‌ها خود از چندین زیرمؤلفه (اجزای کوچک‌تر) تشکیل شده است، امنیت در تمام زیرمؤلفه‌ها منجر به برقراری امنیت معماری جویش‌گر می‌شود و از دید کاربران یک جویش‌گر امن، جویش‌گری است که ۱- نتایج درست، به‌روزشده و مرتبط ارائه کند ۲- از ارائه وب‌گاه‌های آلوده و هرزنامه در نتایج جستجو جلوگیری کند ۳- هشدارها و پیغام‌های امنیتی را به کاربران قبل از مشاهده وب‌گاه‌های آلوده ارائه کند. بنابراین در این مقاله موضوع‌ها و محورهایی مورد بررسی قرار گرفت که پیاده‌سازی و اعمال آنها در جویش‌گرها، محبوبیت و اعتماد را از دید کاربران برای جویش‌گرها به همراه دارد.

۶- مراجع

[1] P. Hunt and S. Hansman, "A Taxonomy of Network and Computer Attack," Computers and Security, pp.31-43, November 2003.

[2] T. Y and G. A, "Web Search Engines: Practice and Experience," Computing Handbook, 2013.

[1] Poisoning

[2] Online

[3] W. Croft, D. Metzler and T. Strohman, Search engines: Information retrieval in practice, United States: Addison-Wesley, Pearson Education, Inc,2015.

[4] T. Y, G. A, “Web Search Engines: Practice and Experience,” Computing Handbook, 2013.

[5] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine", Computer networks, pp.3825-3833, 2012.

[6] D. Thies, “Google Proxy Hacking: How A Third Party Can Remove Your Site from Google SERPs,” Vancouver, Canada, 2007.

[7] "google-bots-doing sql-injection-attacks," [Online]. Available: www.blog.sucuri.net/2013/11/google-bots-doing sql-injection-attacks.html.

[8] O. Christopher and M. Najork, "Web crawling," Foundations and Trends in Information Retrieval, pp. 175-246, 2010.

[۹] زارع بیدکی، علی محمد، طراحی و پیاده سازی سامانه‌های خزش و رتبه‌بندی اسناد فارسی وب و پیاده‌سازی یک جویشگر فارسی، ۸۹۳۲۲۲۴۱۲، پژوهشگاه فضای مجازی، پژوهشکده فناوری اطلاعات ۱۳۹۱/۱۰/۲۴.

[9] A.M. Zare Bidoki, Designing and Implementing Crawl Systems and Ranking Persian Web Documents and Implementing a Persian Search Engine, 893222412, CyberSecurity Research Institute, IT Research Institute, 2013.

[10] F. Dennis, N. Craswell and V. Vinay, "The impact of crawl policy on web search effectiveness." Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009.

[11] "Web_crawler," [Online]. Available: www.en.wikipedia.org/wiki/Web_crawler.

[12] "Apache Solr: List of security vulnerability-es," [Online]. Availab-le:www.cvedetail-s.com/vulnerability–list/vendor_id-45/product_id-18263/Apache-Solr.html.

[13] "Elasticsearch: Security vulnerabilities," [Online]. Available:www.cvedetails.com/vulnerabilitylist/vendor_id13554/Elasticsearch.html.

[14] L. Jerri, Search Engine Optimization Bible, John Wiley & Sons,2009.

[15] L. Theo, M. Brady and T. Masevic, “A Risk Assessment Method for Negative SEO Attacks using a Game Theoretic Approach,” in Professional Communication Conference (IPCC), IEEE International, 2015.

[16] E. Shmueli, R. Waisenberg, Y. Elovici and E. Gudes, "Designing secure indexes for encrypted databases," Data and Applications Security XIX. Springer Berlin Heidelberg, pp. 54-68, 2005 .

[17] P. Noiumkar و T. Chomsiri, “A Comparison the Level of Security on Top 5 Open Source NoSQL Databases,” Information and Communication Technology in the Faculty of Informatics of the Mahasarakham University, Thailand, 2014.

[18] P. Niesten, "Legal status of search engine result manipulation," Tilburg Institute for Law, Technology, and Society (TILT), Netherlands,2012.

[19] "doorway-page” [Online]. Available: www.moz.com/ugc/protect-your-website-from-becoming-a-doorway-page-google-patent-style.",2015.

[20] "search_engine_security" [Online]. Available: www.pcworld.com/article/232224/search_engine_security.html.

[21] Halfond William G., Jeremy Viegas, and Alessandro Orso. "A classification of SQL-injection attacks and countermeasures." Proceedings of the IEEE International Symposium on Secure Software Engineering. Vol. 1. IEEE,2006.

[22] Aryal, D and Shakya, A. "A Taxonomy of SQL Injection Defense Techniques", Sweden (2011).

[۲۳] حسن کوشککی، شقایق نادری"امنیت در مؤلفه‌ها و معماری جویش‌گرها" کنفرانس بین المللی مهندسی کامپیوتر و فناوری اطلاعات، تهران، دانشگاه تهران،۱۳۹۵/۰۳/۱۲.

[23] H. Kooshkaki, S. Naderi, "Security in Components and Architecture Search Engines." International Conference on Computer Engineering and Information Technology, Tehran, Tehran University, 2016.

[24] "FAST ESP Services - Search Technologies" [Online]. Available: www.searchtechnologi-es.com

[25] "searchengineland" [Online]. Available: www.searchengineland.com.

[26] "Search Engine Security" [Online]. Available: www.blog.sucuri.net

[27] "Spamdexing" [Online]. Available: www.en.wikipedia.org/wiki/Spamdexing

[28] "Enterprise Search" [Online]. Available: www.ideaeng.com/security-eprise-search-p1-0304

[29] "Hadoop Framework" [Online]. Available: www.hadoop.apache.org.

[30] "Heuristic Algorithm" [Online]. Available: www.en.wikipedia.org/wiki/Heuristic_(computer_science).

[31]"penguingoogle"[Online].Available:www.mashable.com/2012/07/03/penguin-google-seo/#fGTDYX2hNaqX.

[32] "How Search Works" [Online]. Available: www.google.com/insidesearch/howsearchworks/thestory.

[33] "How Search Works" [Online]. Available: www.support.google.com/webmasters/topic/6001971?hl=en&ref_topic=6001981.

حسن کوشککی مدرک کارشناسی خود را در رشته مهندسی فناوری اطلاعات در سال ۱۳۹۱ از دانشگاه پیام نور و مدرک کارشناسی ارشد خود را در رشته مهندسی فناوری اطلاعات-امنیت اطلاعات در سال ۱۳۹۴ از دانشگاه تربیت مدرس دریافت کرد. زمینه پژوهشی مورد علاقه ایشان امنیت شبکه و اطلاعات، شبکه‌های کامپیوتری، امنیت مالتی مدیا، جریان‌سازی مدیا روی اینترنت، جریان‌سازی ویدیو نظیر به نظیر، امنیت سامانه‌های مبتنی بر شهرت و طراحی امن معماری شبکه است.

شقایق نادری کارشناسی خود را در مهندسی کامپیوتر از دانشگاه خوارزمی در سال ۱۳۷۹ دریافت کرد. تحصیلات کارشناسی ارشد و دکترای خود را در رشته مهندسی کامپیوتر گرایش نرم‌افزار در دانشگاه تربیت مدرس به‌ترتیب در سال‌های ۱۳۸۱ و ۱۳۹۱ به پایان رسانده است. او در حال حاضر استادیار پژوهشگاه ارتباطات و فناوری اطلاعات ایران (مرکز تحقیقات مخابرات ایران) است. زمینه‌های پژوهشی مورد علاقه او درک تصویر، یادگیری ماشین، و امنیت وب است.

نسرین تاج مدرک کارشناسی خود را در رشته مهندسی کامپیوتر گرایش نرم‌افزار از دانشگاه آزاد اسلامی واحد تهران جنوب دریافت و همچنین کارشناسی ارشد خود را از دانشگاه علم و صنعت ایران در رشته مهندسی فناوری ارتباطات و اطلاعات گرایش مخابرات امن دریافت کرد. ایشان هم‌اکنون دانشجوی دکترای DBA دانشگاه تهران است. زمینه‌های پژوهشی ایشان امنیت سامانه و اطلاعات، محرمانگی هم‌چنین مدیریت استراتژیک و مدل‌های پیشرفته کسب و کار است. وی تاکنون مجری و ناظر چندین پروژه پژوهشی در پژوهشگاه فناوری ارتباطات و اطلاعات بوده است.

مهسا امیدوار سرکندی مدرک کارشناسی خود را در رشته مهندسی کامپیوتر گرایش نرم‌افزار در سال ۱۳۹۳ از دانشگاه آزاد اسلامی واحد تهران جنوب دریافت کرد . وی هم‌اکنون دانشجوی کارشناسی ارشد رشته نرم‌افزار دانشگاه صنعتی امیرکبیر است. زمینه‌های پژوهشی مورد علاقه ایشان امنیت نرم افزار، ارزیابی آسیب‌پذیری و تهدیدهای برنامه‌های کاربردی، تحلیل و طراحی و توسعه نرم افزارهای امن و پایگاهه داده‌های ابری است.