کنترل و حذف سرریزی بافر در برنامه‌ها

برنامه‌نویسی به سبک پترس

امیربهاالدین سبط‌الشیخ

یکی از مشکلات امنیتی که در بعضی از نرم‌افزارها مشاهده می‌شود، سرریزی‌بافر یا Buffer Overflow است. سرریزی بافر عموما در زمان اجرا و بسته به ورودی‌های مختلف برنامه رخ می‌دهد و حتی می‌تواند به هکرها کمک کند تا برنامه را تغییر داده و کدهای مخرب در آن وارد کنند. در این مقاله قصد داریم این موضوع را بررسی کنیم و ببینیم سرریزی بافر چیست و چه‌زمانی اتفاق می‌افتد؟

بافر چیست؟

بافر، حافظه موقتی است که به‌صورت نرم‌افزاری و سخت‌افزاری پیاده‌سازی می‌شود، در روش سخت‌افزاری مثل بافرکیبورد، اطلاعات کلیدهایی که شما روی صفحه کلید فشار داده‌اید در جایی ذخیره می‌شود و سیستم‌عامل آنها را می‌خواند و پردازش می‌کند. بافر نرم‌افزاری به دو صورت قابل پیاده‌سازی است.

۱ـ در سطح سیستم‌عامل: سیستم‌عامل مقداری از حافظه را جهت بافرکردن به‌خوداختصاص می‌دهد. این حافظه برای یکسان‌سازی سرعت دیسک‌سخت و پردازشگر استفاده می‌شود.

۲ـ در سطح نرم‌افزار کاربردی: این حافظه را ما به‌عنوان برنامه‌نویس تعریف می‌کنیم تا امور مختلفی را انجام دهیم.

قطعه‌کد زیر را در نظر بگیرید:

```
void overflow_function (char *str){
   char buffer[10];
   strcpy(buffer, str);}
int main(){
   char big_string[14];
strcpy(big_string,"BufferOverflow");
   overflow_function(big_string);
   return 0;
}
```

از لحاظ قواعد گرامری این قطعه‌کد درست است و باید بدون اشکال‌کار کند اما در زمان اجرا، خطای در حال اجرا صادر خواهد شد با این مضمون:

Stack around the variable buffer was corrupted

این خطا به‌ما می‌گوید که متغیر بافر خراب شده است. بیایید ببینیم این کد دقیقا چه کار می‌کند: ابتدا یک متغیر از نوع رشته به طول ۱۴کاراکتر تعریف کرده و مقدار BufferOverflow را با استفاده از دستور strcpy در آن کپی می‌کنیم. تا اینجا مشکل خاصی نیست و کد درست است. حال مقدار big_string به تابع overflow_function پاس داده می‌شود. این تابع سعی دارد مقدار big_string که به‌آن داده شده است را در buffer کپی کند اما این عمل ناموفق می‌ماند و در نتیجه برنامه خطا می‌دهد. چرا خطا می‌دهد؟

این تابع می‌خواهد مقدار ۱۴کاراکتر را در یک رشته ۱۰کاراکتری کپی کند، طبیعی است که این کار را نمی‌تواند درست انجام دهد، برای همین خطا صادر می‌شود. کمی دقیق‌تر به مساله نگاه می‌کنیم: وقتی تابع overflow_function فراخوانی می‌شود آدرس برگشت به تابع فراخواننده در ثبات SP قرار می‌گیرد. سپس این تابع سعی می‌کند مقدار ۱۴کاراکتر را در ۱۰کاراکتر کپی کند و این عمل با موفقیت انجام می‌شود. فرض کنید بافر در ۱۰خانه اول بعد از آدرس 100H قرار داد و ثبات SP مقدار خانه 10DH را به‌عنوان آدرس برگشتی تابع overflow_function در خود دارد. حال مقدار ۱۴کاراکتر در متغیر بافر، کپی می‌شود و در نتیجه خانه‌های 100H تا 10EH بازنویسی می‌شوند و سپس کار تابع به پایان می‌رسد و سیستم‌عامل قصد دارد با استفاده از آدرس ذخیره شده در SP به فراخواننده تابع overflow_function بر گردد اما از آن‌جایی که آدرس 10DH بازنویسی شده است، پردازشگر نمی‌تواند دستوری را اجرا کند و خطای BufferOverflow صادر می‌شود. خب، هکرها با استفاده از همین خطا به سیستم‌های دیگران حمله می‌کنند. آنها یکسری دستورات به زبان اسمبلی می‌نویسند که در اصطلاح به آنها ShellCode یا Exploit گفته می‌شود.

به‌مثال بالا برمی‌گردیم، فرض کنید به‌جای مقدار "BufferOverflow" یک شل کد به تابع overflow_function داده شود. وقتی کار تابع تمام‌شد، پردازشگر به‌آدرس 10DH می‌رود. مقدار این آدرس دیگر یک مقدار نامعتبر نیست بلکه به یک قطعه‌کد اشاره دارد و پردازشگر، آن قطعه‌کد را اجرا می‌کند و سبب می‌شود برنامه اصلی، کار خودش را درست انجام ندهد. این یک روش برای سوءاستفاده از سرریزی است. روش دیگر دسترسی به بخش‌های محرمانه حافظه است که اطلاعات اساسی سیستم در آن قرار دارد. هکر با تزریق کد خود به برنامه می‌تواند به بخش‌های محرمانه حافظه دسترسی پیدا کند و اطلاعات حیاتی سیستم را مورد سوءاستفاده قرار دهد.

چگونه جلوی سرریزی را بگیریم؟

تنها کاری که لازم است انجام دهیم، بررسی مقادیر ورودی برنامه است تا دقیقا مطابق با اندازه متغیر داده‌ها درون آن ریخته شوند. به‌طور مثال: کد بالا را با تغییر کوچکی اصلاح می‌کنیم و از سرریزی بافر جلوگیری می‌کنیم. کد بازنویسی‌شده به‌صورت زیر خواهد بود:

```
void overflow_function (char *str){
   char buffer[10];
   strncpy(buffer, str,10);}
int main(){
   char big_string[14];
cpy(big_string,"BufferOverflow",14);
   strn
   overflow_function(big_string);
   return 0;
}
```

عکس: bildbevis.se

پرسش و پاسخ

درست‌می‌شود که فقط از طریق آن می‌توان وصل شد. در ضمن فایل‌های مخفی ایجاد می‌شود که متاسفانه به‌وسیله ویروس‌کش هم قابل پاک شدن نیست. سیستم من اکس‌پی و آنتی‌ویروسم نود۳۲ آپدیت ۲۰۱۰/۱/۱۶ است. فایل‌های مخفی به‌شرح زیر هستند:

deep.dream.plain.kalba.zolander.sin

دوست عزیز، سیستم شما آلوده به بدافزاری شده است که به همین دلیل بعضی از ویروس‌یاب‌ها قادر به شناسایی آن نیستند.

برای غیرفعال کردن آن، ابتدا تمام Connection‌های خود را حذف کنید و سپس یک Connection با تنظیمات خودتان ولی با همین نام یعنی z-connection ایجاد کنید.

به این ترتیب، ویروس غیرفعال می‌شود و شما به وسیله همین Connection می‌توانید به اینترنت وصل شوید. برخی از آنتی‌ویروس‌های به‌روز مانند مک‌آفی یا Anti-Virus PLUS قادر به شناسایی و از بین بردن این بدافزار هستند.

پس از غیرفعال کردن این ویروس سیستم خود را با یکی از این آنتی‌ویروس‌ها اسکن کنید. برای پاک‌کردن فایل‌های مخفی نیز پس از غیرفعال کردن بدافزار، سیستم را در حالت Safe Mode راه‌اندازی کرده و آنها را پاک کنید.

دلیل وجود ضربدر قرمز رنگ بر روی آیکون شبکه در نوار وظیفه چیست؟

قطع ارتباط شبکه که می‌تواند دلایل مختلف نرم‌افزاری یا سخت‌افزاری داشته باشد.

یدالله‌شریفی ـ مدتی است که یک‌آی‌دی درست‌کرده‌ام ولی پسوردآن را فراموش کرده‌ام و نمی‌خواهم‌آی‌دی دیگری درست‌کنم.

در هنگام ثبت‌نام‌آی‌دی از شما سوالات امنیتی پرسیده می‌شود تا در صورت فراموشی پسورد، با دادن جواب‌آن سوالات پسورد خود را دریافت کنید. بسته به نوع ارائه‌دهنده صندوق پست‌الکترونیکی، در صورت فراموشی پسورد، آن سوالات از شما پرسیده خواهد شد و اگر شما پاسخ درستی به آنها بدهید، پسورد جدید به نشانی ایمیل دیگری که معرفی کرده‌اید، ارسال خواهد شد.

مجدزاده ـ وقتی سیستم بالا می‌آید با خطای زیر مواجه می‌شوم، علتش چه می‌تواند باشد؟

This application has failed to start because unrar.dll was not found. re-installing the application may fix this problem.

علت آن می‌تواند وجود اشکال در یکی از فایل‌های dll نرم‌افزاری مانند unrar باشد. برای برطرف کردن این مشکل طبق راهنمای خود سیستم، این نرم‌افزار را دوباره نصب کنید. در صورتی که به‌آن نیاز ندارید می‌توانید آن را Uninstall کنید. برای اطمینان، سیستم خود را با آنتی‌ویروسی قوی و به‌روز اسکن کنید.

علیرضا عبداللهی ـ می‌خواستم ببینم فایل‌هایی که با فرمت utx هستند را با چه نرم‌افزاری می‌توان بازکرد؟

این قالب، ویژه بازی بوده و توسط خود بازی هم قابل استفاده هستند. این فایل‌ها برای طراحی زمینه یا شخصیت افراد یا استفاده در بازی‌های رقابتی و دارای تورنمنت استفاده می‌شوند.

کیانا کرمی از قروه کردستان ـ با ارتقای چه قطعاتی می‌توان سرعت رندرینگ و لودکردن فایل‌های حجیم را بالا برد. در واقع چه عواملی دقیقا موثر هستند؟ آیا بدون تغییر در سخت‌افزار می‌توان به‌صورت نرم‌افزاری سرعت‌آنها را بهبود بخشید؟ در BIOS سیستم می‌توان سرعت کلاک و فضای کش سی‌پی‌یو را تغییر داد؟ آیا این خاصیت در سیستم گنجانیده شده است؟

علاوه بر سرعت پردازنده، بالابودن حافظه کارت گرافیک و رم نیز از مهم‌ترین عوامل موثر در افزایش سرعت برنامه‌های گرافیکی هستند. ضمن اینکه دیگر عوامل افزاینده سرعت سیستم نیز به افزایش سرعت در این بخش کمک می‌کنند.

برای لودکردن فایل‌های حجیم نیز عواملی که ذکر شد شامل سرعت CPU و سرعت رم بیشترین تاثیر را دارند. برای افزایش سرعت سیستم به‌صورت نرم‌افزاری می‌توانید از ابزارهای خود ویندوز مانند Scan Disk و Disk Defragmenter استفاده کنید که تاثیر مثبتی برکارایی سیستم دارند.

ضمن اینکه نرم‌افزارهای بهینه‌سازی مانند Tune Up Utilities کمک فراوانی می‌کنند. تغییر سرعت کلاک و کش سی‌پی‌یو تنها در مادربوردهایی امکان‌پذیر است که ظرفیت انجام چنین کاری را داشته‌باشند.