

چگونگی کراک کردن ویندوز، یک راه حل

... من در حال حاضر مشغول کراک کردن جدیدترین کاربردهای، ویندوز ۹۵ هستم. من سعی

میکنم به موقع اصلاحات لازم را در مورد ویندوز قبلی ۳/۱ و یافته‌های ویندوز NT انجام دهم.

اولین چیزی که باید به خاطر بسپارید:

ساختار NE، به هر ویندوز قابلیت اجرایی معادل جدول نمادهای رفع عیب کردن میدهد:

یک نمونه رمزگشا:

اشکال زدایی غیر موثق

یکی از خصوصیات عمده ویندوز بر پایه‌های غیرموفق «توانایی اشکال زدایی» می‌باشد.

یک کلمه در مورد کارکرد غیرموثق در سیستمهای عملی بر پایه MS میکروسافت از قوانین و برتری

سیستمهای عملی در سیستمهای جدید روز (MS-DOS ویندوز، ویندوز ۹۵) با دو هدف پلید و

شیطانی عمده به بهترین نحو بهره‌برداری می‌کند.

(۱) بدست آوردن توافق کامل ورشکسته (یعنی اینکه قلمرو تمام کاربردهای عملکردی

غیرموثق و تغییر آنها به محض اینکه تقارن از آنها استفاده کند). درگیری و تضاد در برابر

بورلند به همین روش انجام شد.

۲) به دست آوردن تمام «برنامه‌ریزهای» آینده به منظور استفاده از ویندوز به عنوان یک «جعبه خالی» که تنها مهندسیین میکروسافت (اگر وجود داشته باشند!) میتوانند به آن تسلط پیدا کنند. بنابراین هر کسی قادر خواهد بود از این افتضاح جا نیافته از میکروسافت جرعه‌ای بنوشد بدون اینکه شانس برای تغییر یا اصلاح آنها داشته باشد.

۳) همانگونه که عجیب به نظر میرسد، تنها مجمع رمزگشای عالی قادر به رویارویی با این طرحهای تحمل ناپذیر است. تمام دولتهای کودن و واسطه‌های سیاسی - در مقابل آن - در پشت سرپوش «بازار» «آزادی» به منظور اجازه دادن و توسعه دادن به چنین پیشرفتهای شدیرانه‌ای، خود را پنهان کرده‌اند (من صحبت میکنم به طوریکه اگر آنها قادر به رویایی با چنین اهداف و برنامه‌هایی بودند، حتی اگر خواهان چنین کاری بوند، آنها این کارها را نمیکردند).

مطمئن باشید، آنها برای دولتها عمداً به منظور کار کردن برای تماشاچیان و سیار زالوصفتها، و دلالان سیاسی که سپری در برابر فتودالیسم هستند، ایجاد شده‌اند:

شما میتوانید «دموکراسی» را فراموش کنید، تنها قانون موجود حکومت الیگارشی (ثروتمندان) شدیداً بر اساس پول، ارتباطات شخصی، بی فرهنگی، عدم وجود دانش و دیکتاتوری همراه با اختلاق تند از طریق تلویزیون به منظرو رام کردن برده‌ها، تا اینجا کافی است)..

موقعیت ویندوز در برخی از موارد شبیه موقعیتهای قبلی در DOS است. در جاییکه برای سالها کلید «ذخیره کن اما اجرا نکن» توسط عیب زدها، از قبیل [عیب زدایی]، [سیمدب]، و [مرور رمز] به وسیله میکروسافت محفوظ نگه داشته شد.

کتابخانه عیب زدایی ویندوز، WINDEBUG.DLL تعدادی از عملکردهای غیر موثق و حتی و ربطهایی را که ایجاد می‌کند، غیر موثق هستند! عملکرد WIN DEBUG () توسط تمام

عیب‌زدهای ویندوز در دسترس مورد استفاده قرار می‌گیرد که اینها شامل [CVW] (مرور رمز برای ویندوز). [IDW] (عیب زدای توریو برای ویندوز)، [چند دامنه] و [C سریع برای ویندوز] (دوتای آخر GUI هستند، یعنی عیب‌زدهای متن نیستند. استفاده از WINDEBUB () در خروجی MAPWIN از ارتباط عیب‌زدهای COZ به آن در زمان حرکت از طریق عملکرد حیرت‌انگیز () GETPROC ADDRESS نشان داده شده است.

WINDEBUB() یک نوع ۳۲ بیتی خرد شده است، برای ویندوزهای قدیمی ۳، از عملکرد () DOSPFTRACE با سندیت ضعیف از OS/2 ۱.x است (این سیستم‌های عملی قبلی را مطالعه کنید! با مطالعه گذشته شما همه چیز را متوجه شوید! گاهی اوقات من فکر می‌کنم که تنها راه برای تفکیک و رمزگشایی صحیح این است که بیشتر یک تاریخ‌نگار نرم‌افزار باشیم تا یک برنامه‌نویس. : FAC SAPIAS ET LIBER ERIS، DOSPTRACE، به نوبه خود، بر پایه () PTRACE در یونیکس استوار است.

شبهه به () DOSPTRACE، () WINDEBUB نیز دستورهایی از قبیل (برو یا گام - ساده، فهرست اسامی را بخوانید و بنویسید، حافظه را بنویسید و بخوانید) را صادر می‌کند. این سیستم به ملاقات‌کننده خو حتی هنگامیکه دستور کامل شده یا هنگامیکه یک وقفه ایجاد می‌شود (یا یک ذخیره DLL) برمیگردد. این دستورها و اختارها در یک ساختار بزرگ که آدرس آن در WINDEBUB وارد شده، ظاهر می‌شود.

WINDEBUB() تحت عنوان CVWIN.DLL (و TDWIN.DLL) برای ویندوز ۳/۱، نام گرفت. تمام رمزگشاهای بایستی آن را مطالعه کرده و بالاترین سند ممکن را در مورد آن بدست آورد. همانطور که در زیر می‌بینید، مطالعه TOOL.HELP.DLL (آنچه که میکروسافت می‌خواهد تا شما را با آن سرگرم کند) و INT-41h (ربط عیب‌زدای واقعی)، ارزشمند است.

وقفه در کنترل تحت ویندوز

وقفه در کنترل تحت ویندوز، می‌تواند مشکل آفرین باشد. شما باید از Tool help (یک لوبرتوی هراس انگیز برای برنامه شما) یا داشتن یک رمز مخصوص برای استاندارد از طریق روشهای پیشرفته، به خاطر اینکه اطلاعات در یک دسته وقفه یا کنترل کننده بین دو روش ویندوز تفاوت قائل است به علاوه، بعضی از کنترل کننده‌ها با استفاده از INT-21h نصب شده‌اند. و این در حالی است که سایر آنها با استفاده از سرویسهای DPM1 نصب شده‌اند. Tool help دارای شمار اندکی از رمزهای داخلی است که وقفه‌ها را به نحوی پخته و دست کاری کرده که به شکل قابل هضمی درآمده‌اند.

به خاطر داشته باشید که ویندوز از خطاهای GP به عنوان یک متد «دزدی کامپیوترهای» از طریق انتقال تلفنی که ضوابط قانونی ۸۰X۸۶ مجاز نیست، استفاده می‌کند. سیستم حافظه مجازی مد پیشرفته، از طریق خطای صفحه به اجرا درمی‌آید.

برخی از ابزار برای رمزگشایی ویندوز (به درس ۹ نگاه کنید)

عیب زداها

CVW و TDW (شما باید اجزاء عملکردی را بشناسید، آدرس را از قبل به منظور رمزگشایی یک کارکرد تنظیم کنید).

WCB [codeback ویندوز] توسط Le slie pusztai

WDEB 386 میکروسافت (ناقص بوده، و احتیاج به مانیتور دیگری دارد)

ویندوز SOFT-1Ce

بهترین عیب زدای ویندوز! [که تاکنون شناخته نشده است!] NuMega آنقدر عالی است که من از

رمزگشایی محصولات آنها متأسفم!

[Winice] مطلقاً اساسی ترین عیب یاب بوده و دارای مزایایی برای رمزگشاهای ویندوز است. آن

را تهیه کنید!

بازرسان بعد از مرگ

(بسیاری از اشیاء و اجناس مشترک)

قدیمی و خراب

بازپرس ویژه رگهای غیر طبیعی، غیره

بهترین! دارای خاصیت Buidsym

خانم دکتر واتسون

sym از یک exe بدون اطلاعات عیب یابی فایل

بازرسان ویندوز بورلند

ایجاد می کند.

بازرسان

قدیمی

خانم جاسوس

بهترین نوع، بعدی را انتخاب کنید

وین سایت بورلند

(از جف ریجتر):

ویندوز میکروکوئیل، دمی استیفاء

و ویر (نگهداری انتخاب پیام انتقال)، کنترل، مکانیک و بوم شناس

مداخله گرها

[INFSPY.EXE] بایت ۲۳۱۰۴۲۴، نوع ۲۸/۸/۱۹۹۴، توسط رئیس طراحی نرم افزار. ۲/۰۵

ممکن است که کاملترین نوع باشد.

[supersp4.exe] بایت ۲۴۰۵۷۳، ۱۰/۶/۱۹۹۴، کاملاً دستی برای اطلاعات سریع.

[winview.exe] بایت ۳۰۰۸۳۲، نوع ۳/۰۰ توسط scott Mccraw.

۱۹۹۰-۱۹۹۲ MS(C) این نوع قدیمی MS-SPY است که توسط MS توزیع شده است.

[TPWSPY.EXE] بایت ۹۰۴۷۲. کاملاً ابتدایی، اما شما میتوانید از طریق آن به رمز منبع پاسکال دست یابید.

داخل یک عیب یاب ویندوز ۹۵

شما میتوانید یک برنامه را در یک سطح مونتاژ - زبان بدون داشتن هیچگونه اطلاعاتی از عیب‌زدایی، عیب‌یابی کنید. [عیب‌زدایی] داس همین کار را انجام داده است، به وقفه‌ها و تک‌گام‌ها اجازه داده. و تمام آنها موافقت کرده‌اند که سخت‌افزار باید همراهی کند. اگر به زمان Z80 4-MHZ برگردیم، شما از یک عیب‌زدایی استفاده کرده‌اید که به رمزهای OP وقفه از طریق جریان ساختار متصل بوده تا brak point ایجاد کند.

هیچ چیز تغییر نیافته است. و اینطور چگونگی اشکال‌زدایی بک برنامه بر ۸۰۵۸۶ (=پنتیوم) مطرح شده است. معماری x86 شامل نرم‌افزار وقفه است. کد OP تک‌بایتی XCC، ساختار INT_30 است که برای عیب‌زدها ذخیره شده است. شما میتوانید کد INT_0 3OP در مکان ساختار برنامه کد OP در جایی که وقفه ایجاد می‌شود و جایگزین کد اولیه OP در زمان وقفه می‌شود. در ۸۰۳۸۶ و بعداً. شما میتوانید یک نماد ثبت شده‌ای را که به پردازش گر می‌گویید که یک INT_01 غیرمزامم را برای اجرای هر ساختار ماشین ایجاد کند، تنظیم کنید.

Win 32SDK (ابزار توسعه گر نرم‌افزار ویندوز ۹۵، شامل عملکردی است که به یک برنامه اجازه میدهد تا برنامه دیگری را راه‌اندازی کرده و آن را عیب‌زدایی کند. عیب‌زدای SDK، از API چگونگی مانعها و بردارهای وقفه را کنترل می‌کند. نتایج منطقی چنین روندهایی این است که مردم کمتر و کمتری قادر به شناختن و دانستن آنچه که درون یک ساختار می‌گذرد، نیستند. قسمت اعظم برنامه‌نویسها در ظرف مدت چند سال قادر به ذخیره کاربرد مهندسی نیست، مگر اینکه تعداد ادکی که هنوز قادر به فهم زبان ردیف کرده را می‌فهمند، ابزار را در اختیار می‌گیرند تا کاری انجام

دهند. میکروسافت همانطور که واضح و مبرهن است میخواهد که برنامه‌نویسان از روش «جعبه خالی» برای برنامه‌نویسی استفاده کنند. بدینگونه که اجرای «سلام دنیا» کوچک زیبایی را نوشته و آن را به مهندسين میکروسافت واگذار میکنند.

سند ویندوز ۳۲ به نظر وسیع، تقریباً تجملاتی میرسد. تا اینکه شما شروع به کار جدی و کشف نقصهای آن بکنید و شبیه به این واقعیت که کدهای اشتباهها و خطاها موثق نیستند، و APIهای بیشماری یا به طور نادرست یا ضعیف موثق شده به طوریکه شما وقت زیادی را برای تصحیح آنها باید از دست بدهید. آن چیز که ما به آن به طور قطع نیاز داریم یافتن چند رمز درون میکروسافت است. (شبیه پروموتوس خوب قدیمی) که به طور قاچاقی اسناد واقعی را که مهندسين میکروسافت برای خود ذخیره کرده‌اند به خارج ارسال می‌کند. اگر شما این را بخوانید و برای میکروسافت کار کنید، امکان اینکه برای مافوقهای خود به خاطر بشریت دو دوزه بازی کنید و اطلاعات رمزی را به طور قاچاقی به ما منتقل کنید، در نظر میگیرید.

در ویندوز ۹۵، یک برنامه اشکال یاب، یک برنامه را راه اندازی کرده که این کار، به وسیله عملکرد Create process عیب زدایی شده و یک بحث که این برنامه باید عیب زدایی شود. به آن تخصیص داده شود. سپس برنامه عیب زدا یک حلقه را برای اجرای برنامه وارد برنامه می‌کند.

در بالای حلقه عیب زدا با wait for debug event ارتباط برقرار می‌کند.

هر زمان که برنامه wait for debug event آن را بگرداند، شاخصهایی را که در مورد عیب زدایی برنامه صحبت میکنند، تنظیم کنید. این جایی است که عیب یاب مانعها و استثناهای تک گام را به دام می‌اندازد. Wait for debug event ساختاری را که در میان سایر چیزها در بردارنده آدرسی است که عامل بازدارنده را نشان میدهد.

عیب یاب با `Get Thread Context` به منظور به راه انداختن برنامه عیب‌زدایی ارتباط برقرار می‌کند. این برنامه شامل مضامین ثبت‌کننده‌هاست. عیب یاب می‌تواند، در نتیجه ارتباط متقابل با رمزگشا، این ارزشها و مضمونهای حافظه برنامه و ویروس یاب را اصلاح کند.

ویروس یاب به وسیله ذخیره کد `OP` در ساختاری که متوقف شده و در کد `OP _03 INT` واقع شده در میان خود وقفه‌ای ایجاد می‌کند، این همان مارمالاد قدیمی است. هنگامیکه وقفه ایجاد می‌شود، ویروس یاب کد `OP` اولیه را در حافظه ساختاری برنامه جایگزین می‌کند. و برنامه را در متن ذخیره شده وارد کرده به طوریکه اجرای آن در ساختاری که شکسته شده از سر گرفته می‌شود.

برای یک مرحله‌ای کردن یک برنامه، ویروس یاب یک قطعه را در شاخص ثبت‌کننده متن به طوری تنظیم می‌کند به طوریکه پردازشگر برای هر چرخه ساختار یک `INT _01` ایجاد می‌کند. هنگامیکه یک وقفه ایجاد می‌شود، ویروس یاب سیستم را کنترل می‌کند تا ببیند که آیا آدرس وقفه در شماره خط با رمز جدید است یا خیر. اگر چنین نباشد، ویروس یاب به اجرای خود ادامه می‌دهد. در غیر این صورت، ویروس یاب خط جدید را `IDE` نشان داده و برای رمزگشایی صبر کرده تا اقدامی جهت از سرگیری برنامه صورت دهد.

وقتی که برنامه ویروس یابی شده مسکوت گذاشته شده، ویروس یاب با رمزگشا ارتباط برقرار کرده و یک دستیابی کامل به متن و حافظه برنامه ویروس یابی شده پیدا می‌کند. این وسیله دستیابی به رمزگشا اجازه می‌دهد قسمتهایی از کد را آزمایش و اصلاح کند.

به منظور از سرگیری برنامه ویروس یابی شده. ویروس یاب متن برنامه را توسط ارتباط با `set thread contex` و `continue debug event` دوباره تنظیم می‌کند.

سپس، ویروس یاب به بالای حلقه به منظور برقرار کردن ارتباط محدود با wait for debug event برمیگردد.

برای استخراج اطلاعات ویروس یاب از فایل اجرایی ویندوز ۳۲ شما باید چارچوب آن فایل را بشناسید (بهترین کار، برای تمرین، ذخیره برنامه‌های کوتاه است) فایل اجرایی دارای دو بخش است که در سایر فایل‌های اجرایی یافت نمیشود. Stab و stabstr چقدر زیبا اسامی برای آنها انتخاب شده که منطبق با هدف آنهاست. (nomenest onien). شما آنها را داخل یک جدول با فهرستهایی با طول ثابت که در برگیرنده فهرستهایی برای Text, bss, data, idata است در داخل این بخشها گردآورنده‌ها بخشهای مختلف یک برنامه را قرار میدهند.

چندین ساختار متفاوت برای کدگذاری کردن اطلاعات ویروس یابی در یک فایل اجرایی وجود دارد. ویروس یاب Borland's turbo یکی از آنهاست. Microsofts codeview یکی دیگر است. gun-win32 port از Cygnus ساختار stab است. یک معنی سرواژه‌ای آن «جدول نمادها» است. اگرچه جدول دربردارنده اطلاعاتی فراتر از اطلاعات نمادین محض است.

بخش stab. در یک فایل اجرایی قابل حمل جدولی از فهرستها با طول ثابت است که اطلاعات ویروس‌یابی را در ساختار stab نشان میدهد. بخش stabstr. دارای طولهای متفاوت بوده و رشته‌های با پایانه‌های تهی که در هرکدام نکات فهرست جدول stab. مشاهده می‌شود.

سندسازی برای ساختار stab. در چارچوب متن در [ftp.cygnus.com/pub/gnu-win32](ftp://ftp.cygnus.com/pub/gnu-win32) در دسترس است.

stab. در اکثر ساختارهای مرموز، در بردارنده مواد زیر است: اسامی و خصوصیات انواع درونی و استفاده کننده‌ها، آدرس حافظه‌ای هر نماد در حافظه بیرونی، نشانی شمارنده حافظه هر عملکرد، نشانی شمارنده حافظه جایی که هر بیان استوار و محکمی شروع شده و به پایان میرسد، آدرس

حافظه از شماره‌های خط درون فایل هایی با منبع رمز ، و رهر چیز دیگری که یک ویروس یاب به آن نیاز دارد.

ساختار پیچیده و مرموز است به خاطر اینکه میخواهد از هر زبان منبع رمز حمایت کند. این مسوولیت یک برنامه ویروس است که فهرست های stab را به یک صورت معنی‌دار برای ویروس‌یاب به آن زبانی که ویروس یابی در آن صورت میگیرد، ترجمه کند.

ویندوز ۹۵ به دهها سرویس INT-21 از کد ۳۲ بیت استناد می‌کند که شامل KERNEL. 32. DLL و KRN 32 MUTEX است که ظاهراً دستیابی به مخصوصی از مرکز را کنترل می‌کند. بعضی از نقشها در KERNEL 32 توسط WIN 16 MUTEX متوقف می‌شود. حیت وقتی که میکروسافت مورد آنها را تأیید نکند.

بسیار خوب، من میخواهم به رمزگشایی پردازم، باید چه کار کنم؟

من باید یک رمزگشای ساده ویندوز را به شما نشان دهم، بنابراین بدون WINICE دست یافتن به آن آسان است: اجازه دهید [WinPGP 4.1] را بگیریم . جلو - انتها برای انجام PGP در ویندوز، توسط Geilo من بایستی از Q برای این عقیده کار در زمینه رمزگشایی تشکر کنم.

با استفاده از WCB شما میتوانید به سرعت بفهمید که «تبریک» شماره ثبت شما صحیح است و متأسفم، شماره ثبت شما صحیح نیست. وقفه اطلاعات در شروع وقفه در 36038B8 (به ترتیب در ۳۶۰۳۹۳۷ و ۳۶۰۳۸ D۵) به ۱۳۱۰۸۱B صورت می‌گیرد.

با نگاه در ۱۳۰۰۰ و کدهای زیر را شما با یک فشار (۶۸ D۵۳۸) ۳۸D۵ و یک فشار (۶۸۳۷۳۹) ۳۹۳۷ در ۱۳/۰۶۴D و ۱۳/۰۶۴AE بدست می‌آورید.

راه برای رمزگشایی اکنون باز است. شما فقط بایستی آن را پیدا کرده و با آن ارتباط برقرار کنید.

شما روند دقیق آن را برای این نوع از رمزگشایی WIN در قسمت ۲ و ۳ از درس ۸ یاد بگیرید.

حالا اجازه بدهید نگاهی به این جدول بیندازیم.

```
13.0E88 660FBF46F8 movsx  eax, word ptr [bp-08]
13.0E8D 668946F4  mov   [bp-0C], eax
13.0E91 668B46F4  mov   eax, [bp-0C]
13.0E95 6669C00A000300 imul  eax, 0003000A
13.0E9C 668946F0  mov   [bp-10], eax
13.0EA0 668B4606  mov   eax, [bp+06]
13.0EA4 663B46F0  cmp   eax, [bp-10]
13.0EA8 7505      jne   0EAF      <- beggar_off
13.0EAA B80100   mov   ax, 0001  <- flag 1 = "Right!"
13.0EAD EB04     jmp   0EB3     <- and go on
beggar_off:
13.0EAF 33C0     xor   ax, ax   <- flag 0 = "Nope!"
13.0EB1 EB00     jmp   0EB3     <- and go on
```

من از شما میخواهم که نگاهی به این جدول بیندازید.

این همان سوپ قدیمی است، شما درس ۳ و جدول حفاظتی بازیهای احمقانه DOS قدیمی از

دهه ۸۰ را به خاطر می آورید. اینطور نیست؟ این همان سوپ قدیمی است!

در این کاربردهای «بسیار پیشرفته» و «جدید» ویندوز، در WINPGP نوع 1/4 از سال ۱۹۹۵/۱۹۹۶،

دقیقاً همان نوع از حفاظت به منظور «از بین بردن» رمز عبور به کار برده شد!

الف) ورودی استفاده کننده را با ECHO حافظه مقایسه کنید.

ب) اگر با $AX=0$ معادل نیست از دیگری بگیرید.

ج) اگر $AX=1$ از آن رد بشوید ... چقدر خسته کننده است!

علاوه بر آن به MOV EAX اصلاً توجه نکنید، و EAX از کنار مقایسه حرکت می کند. این یک

موضوع معمولی برای این «اعداد رمزی عبور» است. من «در سالهای پیش» کاربرد کوچکی از

رمزگشا را نوشته ام که آن برای وقفه رمز با یک ۶۶ به عنوان اولین ساختار بایتی که در چهار یا

بیشتر ساختار متوالی تکرار می‌شود، تحقیق کرده‌ام. و آن هنوز به من اجازه می‌دهد که از نصف بیشتر این رمزهای عبور ویندوز در کمتر از ۳ ثانیه رمزگشایی کنم.

ساختار IMUL، اعداد سحرآمیز را ایجاد می‌کند. و اگر شما نگاه نزدیکتری به آمار و ارقام ریاضی در قسمت «حذف» بیندازید، به شما در رمزگشایی جدول آنالوگ به کار برده شده به منظور حفظ «دستیابی فوری» کمک کند.

(= نرم افزار ضعیف زمان (tm) & (c)

حالا شما میتوانید کدهای بالا را به ۱۰۱ روش متفاوت رمزگشایی کنید، بهترین راه احتمالاً JZOEAF یا Je OEAF، به Jne OEAF در OEA8 ۱۳ است. شما تنها یک ۷۴ را در مکان ۷۵ مینویسید، همانطوری که این کار را در مورد (چقدر خسته کننده .. ۱۹۷۸) انجام دادید: این واقعاً همان سوپ قدیمی است! (اما شما چند اثر جدید در درسهای بعدی خواهید دید). بسیار خوب، این هم از این درس، خواننده، هیچ کدام از درسهای آموزشی در مورد WEB نیست.

شما درسهای عقب مانده را میتوانید تنها از طریق پست دریافت کنید. از طریق (via anon. Penet.fi) شما میتوانید آنها را همراه چند اثر دیگر دریافت کنید. من آنها را قبلاً میدانستم. اما اگر آنها واقعاً جدید باشند یک مدرک کامل به شما داده می‌شود. و حتی اگر آنها جدید نباشند، من در مورد شما اینطور قضاوت خواهم کرد که شما آنها را با کار خوند کشف نکرده‌اید یا اینکه شما واقعاً کار پرتلاشی روی آنها انجام داده باشید. من هرگز درسهای باقیمانده را برای شما نخواهم فرستاد. پیشنهادات و انتقادات شما از روند کلی نوشته‌های من با استقبال روبرو می‌شود.