بسمه تعالی

[Subject]

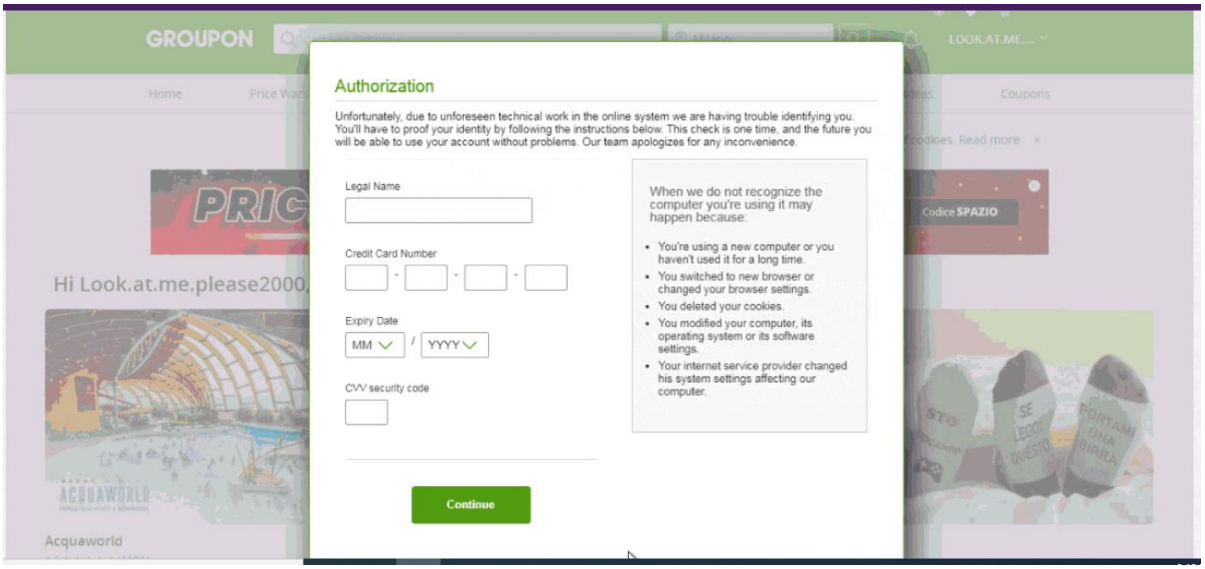
گزارش بدافزار

تروجان قدرتمند بانکی به نام DanaBot که نخستین بار در ماه می سال ۲۰۱۸ کشف شد با تمرکز بر سرویس‌های موسسات مالی فعالیت خود را در اروپا و استرالیا ادامه می‌دهد. DanaBot از زمان ظهور تاکنون رشد قابل توجهی داشته است و مشابه تروجان بانکی Zeus به واسطه‌ی ماژول‌های plug-and-play (که می‌تواند سریعاً تاکیک و اولویت‌ها را تغییر دهد) شهرت یافته است.

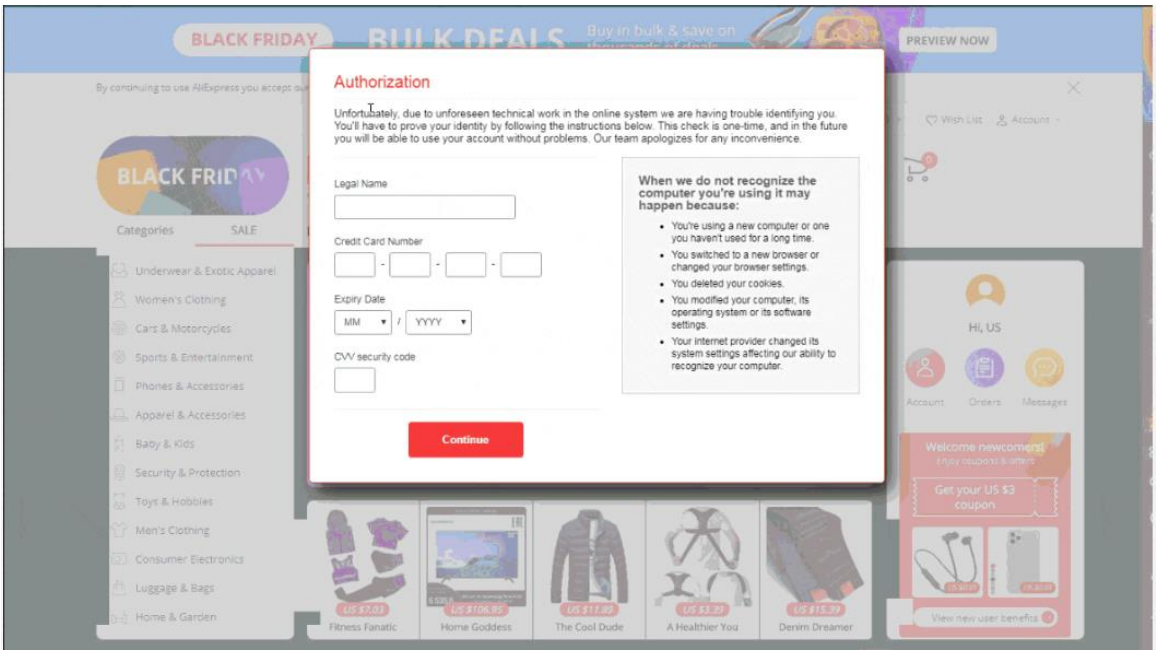
مشابه بسیاری دیگر از تروجان‌های بانکی، DanaBot نیز مرتباً تاکتیک‌های خود را تغییر می‌دهد. این تغییرات ابتدا در سپتامبر سال ۲۰۱۹ مشاهده شد:

- DanaBot در سپتامبر ۲۰۱۹ حوزه فعالیت خود را از موسسات مالی به پلتفرم‌های ecommerce و شبکه‌های اجتماعی تغییر داد؛ این بدافزار ریشه‌های تروجان بانکی خود را ترک نکرده بود بلکه حوزه فعالیت خود را به اهداف جدید توسعه داده بود.
- در ادامه‌ی افزودن اهداف جدید، DanaBot از یک ماژول باج افزار استفاده کرد که نشان از تغییر اولویت‌های این بدافزار دارد.
- DanaBot از دو روش برای انجام این حملات و سرقت استفاده می‌کند:
  - روش اول، ساخت فرم‌های تقلبی در سایت‌های مشهور است. این روش توسط دیگر تروجان‌های معروف بانکی نیز با به کارگیری «کتابخانه جدول جاوا اسکریپت» مورد استفاده قرار گرفته است.
  - روش دوم شامل استفاده از iframe مخرب و سوءاستفاده از فریمورک p.a.c.k.e.r می‌باشد. لازم به ذکر است که p.a.c.k.e.r روشی قانونی برای فشرده‌سازی و درهم‌سازی کد جهت ساخت دستورات و کنترل مکانیسم‌های ارتباطی است.

مطابق شکل شماره ۱ و شکل شماره ۲ به کار گیری این روش‌ها در سایت‌های معروفی مثل AliExpress و Groupon توسط DanaBot مشاهده شده است.



شکل شماره ۱: فرم مخرب در سایت Groupon

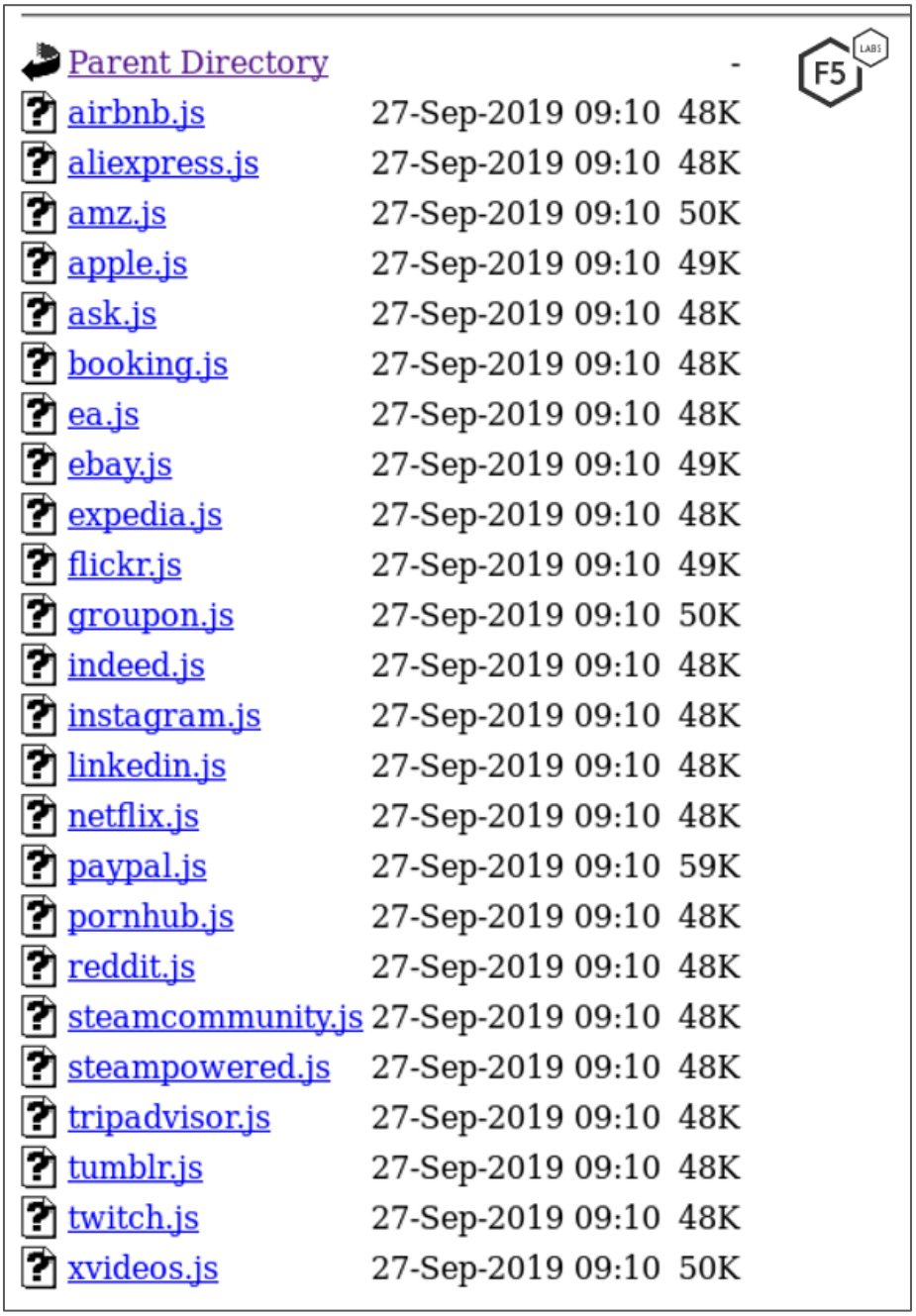


شکل شماره ۲: فرم مخرب در سایت black friday

۱ شکست تکنیکی : جدول‌ها و فرم‌های مخرب

مشابه دیگر تروجان‌های مخرب بانکی مانند Zeus و Goz، DanaBot نیز به دلیل عملکرد خود در تزریق کد در بستر وب (web injection) شناخته می‌شود. در واقع Web injection راهکار اصلی این تروجان در به سرقت بردن پول و اعتبارات (مانند رمز عبور و مشخصات کارت بانکی) قربانی می‌باشد. محققان توانستند با

بررسی سرور DanaBot تحلیل تعدادی از Web injection های سفارشی را آغاز کنند. مکانی که Webinjectهای مذکور قبل از تزریق به هدف نگهداری می‌شوند در شکل شماره ۳ قابل مشاهده است.



شکل شماره ۳: سرور مخرب DanaBot جایی که تمامی webinjectionهای سفارشی قرار دارند

قبل از اینکه فرم مخرب برای کاربر نمایش داده شود کتابخانه‌ی مخرب جدول جاوااسکریپت کار خود را با چک کردن سایتی که کاربر به آن وارد شده و تطابق آن با لیست وب سایت‌های هدف DanaBot شروع می‌کند. سیستم عامل قربانی باید از قبل به بدافزار DanaBot آلوده شده باشد؛ در این صورت بدافزار می‌تواند

چک کند که آیا قربانی به یکی از سایت‌های مورد هدف DanaBot وارد شده است یا خیر؛ این کار با بررسی یک المان ساده HTML صورت می‌گیرد .

```
_tables.start = function () {
    _tables.set('logout', function () {
        if (_tables.id('sign-out')) {
            return true;
        }
        return false;
    });
```

شکل شماره ۴: قطعه کدی که بررسی می‌کند که آیا صفت "sign out" در صفحه وب وجود دارد یا خیر.اگر وجود داشته باشد DanaBot عملیات مخرب خود را آغاز می‌کند

```
if (_tables.findout(document, 'span', 'class:welcome\-title')) {
    return true;
}
return false;
```

شکل شماره ۵: قطعه کدی که چک می‌کند آیا المان «span»ی که صفت کلاس آن (class attribute) شامل «welcome» باشد وجود دارد یا خیر. اگر وجود داشته باشد DanaBot یک فرم تقلبی ایجاد و به کاربر نمایش می‌دهد.

مطابق شکل شماره ۴ و شکل شماره ۵ بدافزار با یک بررسی ساده سورس کد HTML ورود کاربر به سایت را تشخیص می‌دهد. به محض اینکه کد، وجود دو المان HTML را در صفحه وب تشخیص می‌دهد، تولید فرم تقلبی را شروع می‌کند.

بدافزار با استفاده از کتابخانه جدول جاوااسکریپت فرم‌های پرداخت تقلبی که کاربران اطلاعات خود را در آن وارد می‌کنند را تولید می‌کند. درگذشته کتابخانه جدول جاوااسکریپت توسط تروجان‌های بانکی قدرتمندی مانند Zeus و Ursnif/Gozi مورداستفاده قرار می‌گرفتند.

منطق پیاده‌سازی شده سمت کاربر شامل روش‌های زیر است:

- بررسی معتبر بودن تاریخ و یا ایمیل وارد شده توسط کاربر

```
check_day : function (dd) {
    if (parseFloat(dd) > 0 && parseFloat(dd) < 32 && (dd + '').length == 2) {
        return true;
    } else {
        return false;
    }
},

check_month : function (mm) {
    if (parseFloat(mm) > 0 && parseFloat(mm) < 13 && (mm + '').length == 2) {
        return true;
    } else {
        return false;
    }
},

check_email : function (email) {
    var re = /^(([^<>()[\]\\.,;:\s@\"]+(\.[^<>()[\]\\.,;:\s@\"]+)*)|(\".+\"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.
    return re.test(email);
},

check_year : function (yy, format) {
    switch (format) {
    case ('YY'):
        if (parseFloat(yy) >= 15 && (yy + '').length == 2) {
            return true;
        } else {
            return false;
        }
        break;
```

شکل شماره ۶:تابع بررسی اعتبار تاریخ وارد شده

- استفاده از یک تابع decoder برای اشیا HTML
- غیرفعالسازی دکمه submit فرم بااستفاده از event attribute attacher (مجبور کردن کاربر به استفاده از فرم مخرب که دکمه event مخرب دارد)

```
input : function (input, type) {
    switch (type) {
    case ("block"):
        if (input) {
            input.onkeup = function (evt) {
                var evt = (evt) ? evt : ((event) ? event : null);
                var node = (evt.target) ? evt.target : ((evt.srcElement) ? evt.srcElement : null);
                if (evt.keyCode == 13) {
                    if (evt.stopPropagation) {
                        evt.stopPropagation();
                    } else {
                        evt.cancelBubble = true;
                    }
                    return false;
                }
            };
```

شکل شماره ۷: تابع مربوط به غیرفعالسازی دکمه «ورود با استفاده از Keyboard» و جایگزینی عمل «کلیک» مخرب

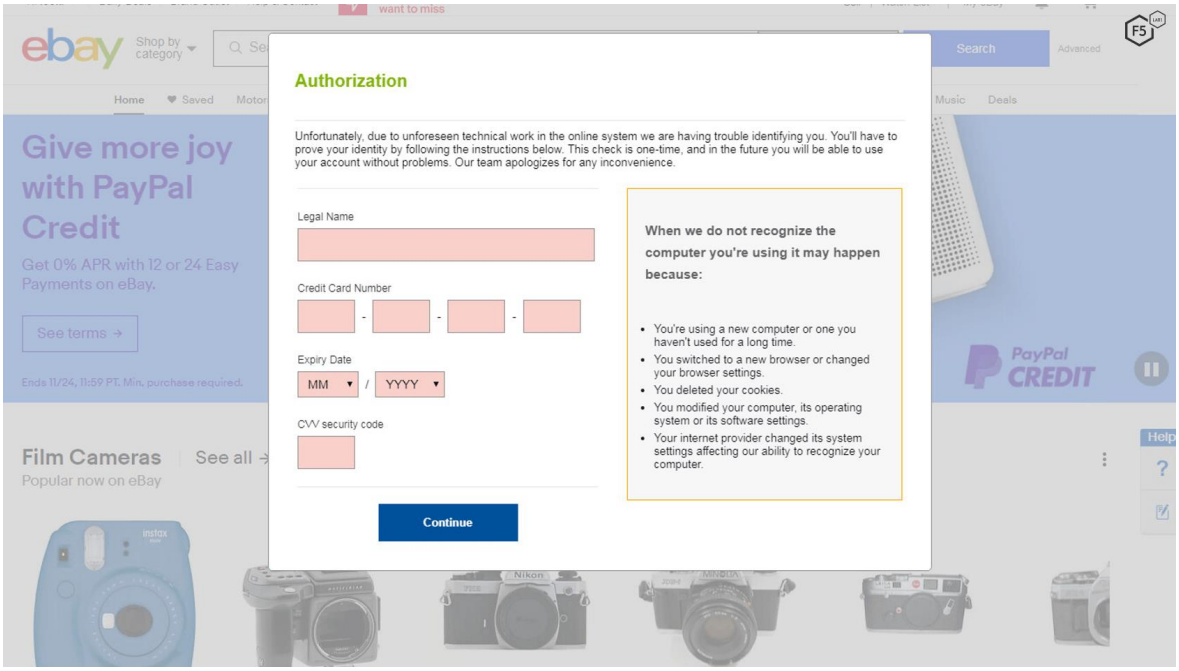
- جایگزینی دکمه‌های submit با دکمه‌های مخرب حاوی عملکرد کلاه بردارانه

```
replacebutton : function (button, func) {
    var newButton = document.createElement(/image/igm.test(button.tagName) ? 'img' : button.tagName);
    for (x in button.attributes) {
        if (notnull(button.attributes[x]) && notnull(button.attributes[x].name) && notnull(button.attributes[x]
            if (button.attributes[x].name == "onclick" ||
                button.attributes[x].name == "name" ||
                button.attributes[x].name == "disabled" ||
                (button.attributes[x].name == "href" && !/image/igm.test(button.tagName)) ||
                button.attributes[x].name == "id") {
                continue;
            }
            if (button.attributes[x].name == "type" && button.attributes[x].value == "submit") {
                newButton.setAttribute('type', 'button');
            } else {
                newButton.setAttribute(button.attributes[x].name, button.attributes[x].value);
            }
```

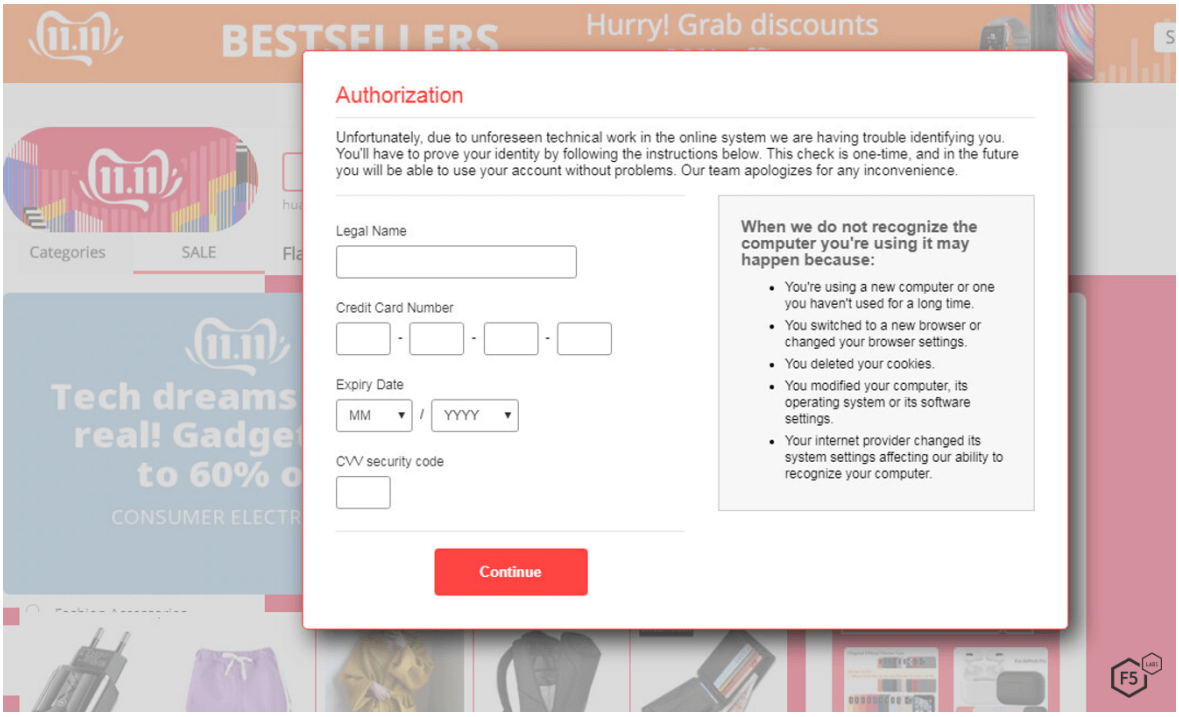
شکل شماره ۸: تابع مربوط به جایگزینی دکمه معتبر با دکمه تقلبی

- استفاده از یک ابزار جاوا اسکریپت برای رمزنگاری/رمزگشایی URLها
- بررسی null بودن متغیرها و objectهای جاوا اسکریپت

پیش از سپتامبر ۲۰۱۹ و ظهور این بدافزار این تاکتیک برای به سرقت بردن اطلاعات بانکی استفاده نشده بود. با این رو DanaBot آشکارا به جمع آوری اعتبارات و اطلاعات بانکی کاربران می‌پردازد و کاربران نیز با اطمینان از اینکه سایتی که در آن هستند معتبر و قانونی است اطلاعات حساس کارت بانکی خود را در فرم تقلبی وارد می‌کنند.



شکل شماره ۹: فرم مخرب DanaBot که در جهت سرقت اطلاعات بانکی در سایت ebay نمایش داده شده است

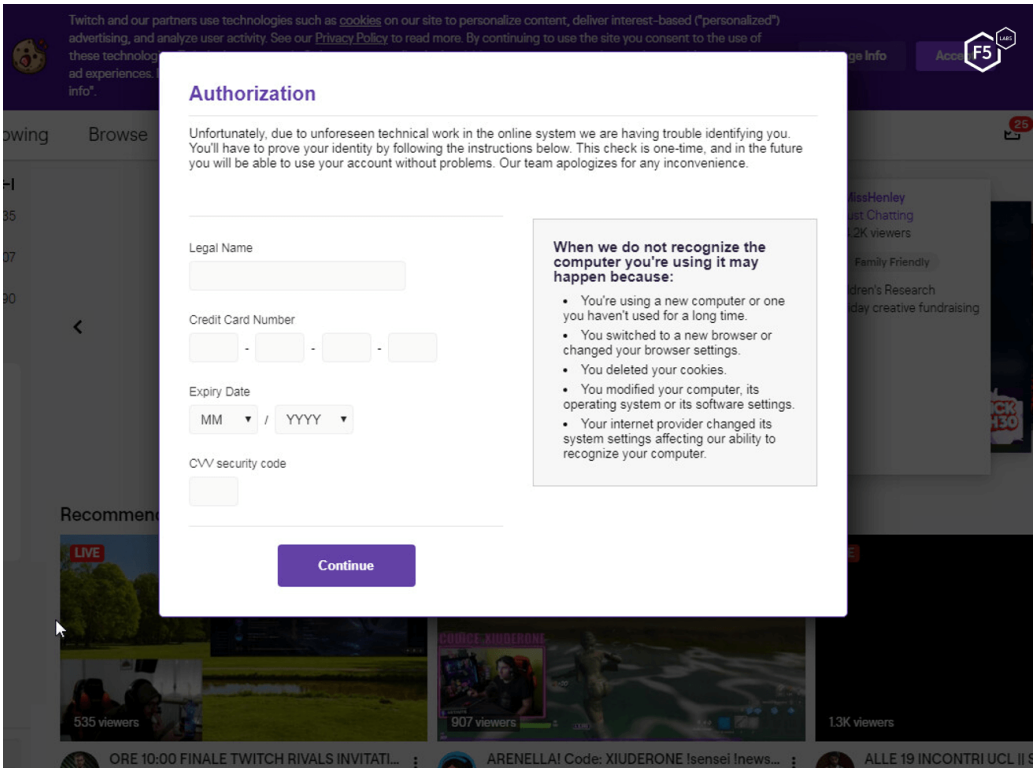


شکل شماره ۱۰: فرم مخرب و تقلبی در سایت AliExpress که یک سایت معروف ecommerce

این تاکتیک مشابه روش‌های بکار گرفته شده در دیگر تروجان‌های بانکی است که در آن کاربر از سرقت اطلاعات و اعتبارات بانکی خود و کلاه‌برداری صورت گرفته اطلاعی ندارند. در این موارد در همه‌ی قسمت‌های این وب‌سایت‌های قانونی کلاه‌برداری صورت نمی‌گیرد بلکه تنها قسمت کوچکی از وب سایت مانند یک فرم popup مخرب خواهد بود درنتیجه قربانی به تقلبی بودن فرم شک نخواهد کرد.

مثال‌های مطرح شده (eBay، AliExpress و Groupon) از آن جهت قابل توجه هستند که روی آوردن بدافزارها از سوءاستفاده از سرویس‌های موسسات مالی به پلتفرم‌های معروف ecommerce (تجارت الکترونیکی) که در آن کاربران به وارد کردن اطلاعات بانکی عادت دارند را نشان می‌دهد.

در کنار اهداف DanaBot در حوزه تجارت الکترونیک (ecommerce)، این بدافزار اهداف خود را به شبکه‌های اجتماعی و وب‌سایت‌های streaming گسترش داده است. این اهداف شامل وب‌سایت برجسته Twitch در حوزه پلتفرمlive streaming برای gamerها نیز می‌شود. در این وب سایت کاربران می‌توانند به تماشای بازی و یا چت با دیگر کاربران برخط بپردازند؛ همچنین می‌توانند اطلاعات کارت بانکی خود را برای خرید Twitch Prime و یا پشتیبانی از کانال‌های به خصوص در وب‌سایت وارد کنند. همانطور که در شکل شماره ۱۱ مشاهده می‌شود بدافزار با به کار گیری تکنیک مشابه بخش قبل (web injection) اطلاعات کاربران را به سرقت می‌برد.



شکل شماره ۱۱: فرم مخرب و تقلبی در سایت Twich برای سرقت اطلاعات

۲ شکست تکنیکی :درهم ریختگی و استفاده از iFrame

در کنار استفاده از تکنیک‌های جدیدی که در بخش‌های قبل به آن اشاره شد،DanaBot در ماه سپتامبر ۲۰۱۹ در حال استفاده‌ی پویا از فریمورک فشرده‌ساز p.a.c.k.e.r مشاهده شد که روشی قانونی برای فشرده‌سازی و درهم‌سازی کد جهت ساختن یک مکانیسم CNC[1] است.

در کنار فشرده ساز p.a.c.k.e.r، تابع «eval» جاوااسکریپت نیز بکار گرفته می‌شود. این تابع به دلیل عدم بررسی مقدار ورودی که به آن پاس داده شده می‌شود، آسیب‌پذیر است و در نتیجه با هر مقدار ورودی اجرا می‌شود. این تابع آسیب‌پذیر یک رشته فشرده شده را که خروجی فشرده ساز p.a.c.k.e.r است به عنوان ورودی دریافت می‌کند. سپس یک اسکریپت تولید می‌شود که login بودن قربانی در سایت را مشخص می‌کند.

[1] Command and Control

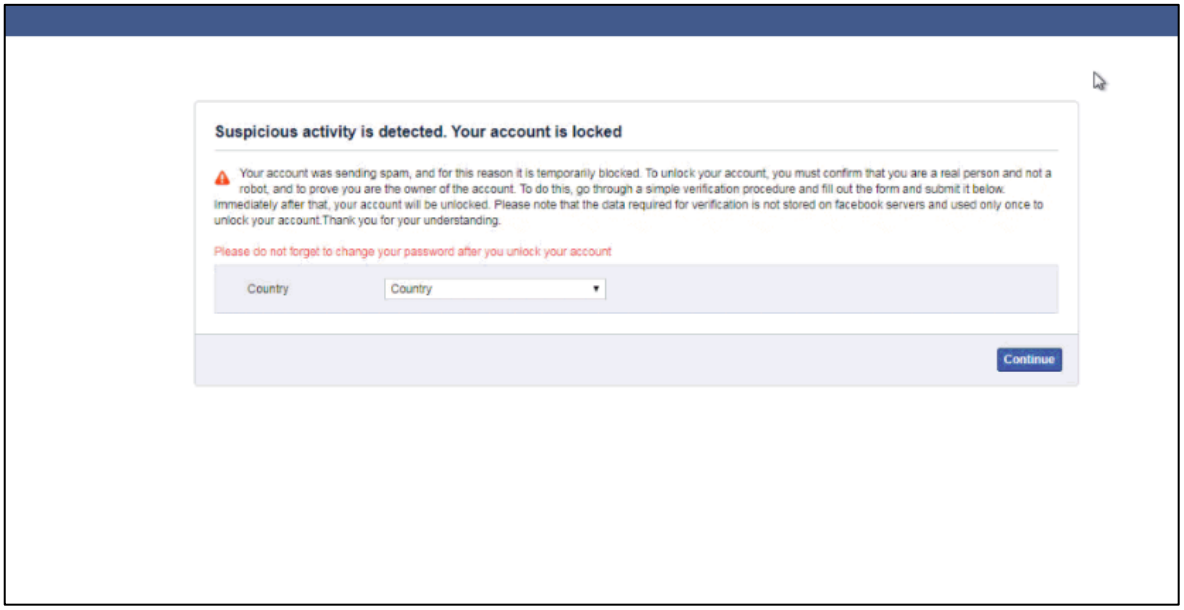
بعد از ساخت اسکریپت، بدافزار از یک iFrame برای ارسال پیام و دریافت پاسخ استفاده می‌کند که با استفاده از مکانیسم postMessage انجام می‌گیرد که ارتباط با پنجره‌ی مادر (Parent Window) در وب‌سایت را ممکن می‌سازد. سپس اسکریپت، پیام‌هایی را که توسط «اسکریپت تولید شده توسط تابع eval» خوانده شده بود دریافت می‌کند.

جریان کامل این عملیات مخرب در زیر آمده است:

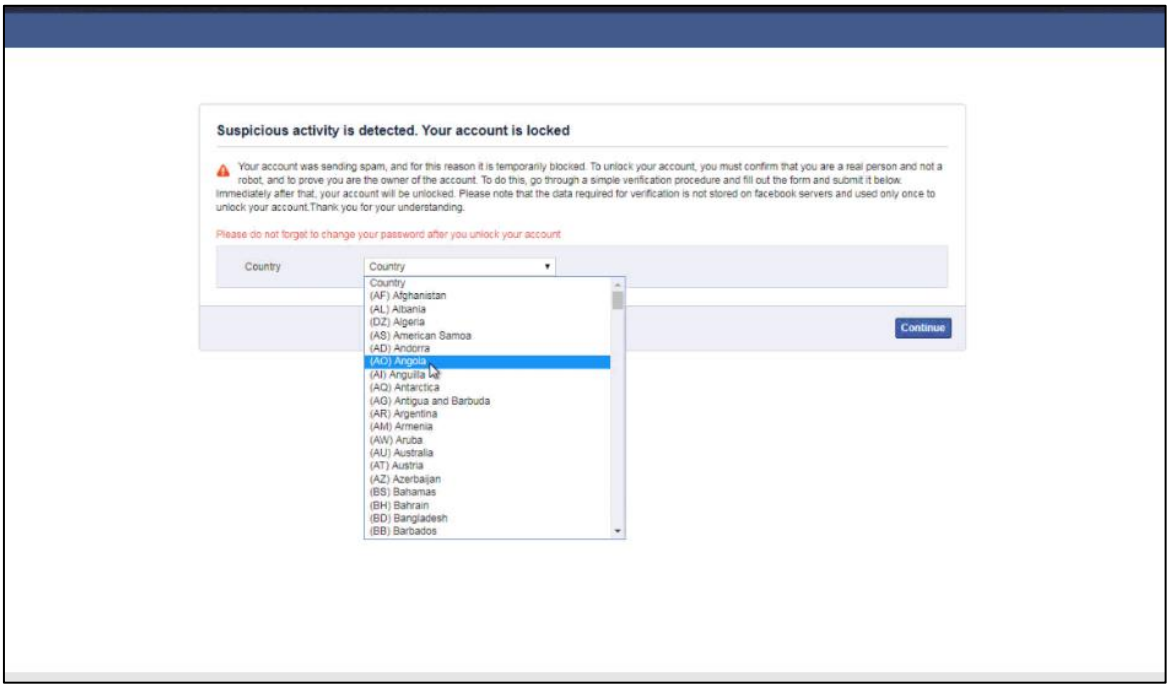
- مهاجم از فشرده ساز p.a.c.k.e.r برای ساخت پویای تابعی به نام NCCVBVGrabLoader استفاده می‌کند. این تابع Login بودن قربانی در سایت را بررسی و ارتباطات با iFrame را کنترل می‌کند (id صفت مربوط به آن «pmiframe» نام دارد). NCCVBVGrabLoader خروجی iFrameها (id=pmiframe) را که از سرور دریافت شده است به عنوان مقدار ورودی دریافت می‌کند.
- پاسخ توسط top.postmessage (که حاوی تابع eval است) دستکاری می‌شود. تابع eval، html مخرب را به وب‌سایت هدف تزریق می‌کند. این عمل یک کنترلر جدید برای ارتباط با iframeها فراهم می‌کند.
- در آخر اسکریپتی تولید می‌شود که لیست کشورها و زبان‌هایی را که فرم باید با آنها پر شود، دریافت می‌کند.

- **.Error! Reference source not found**

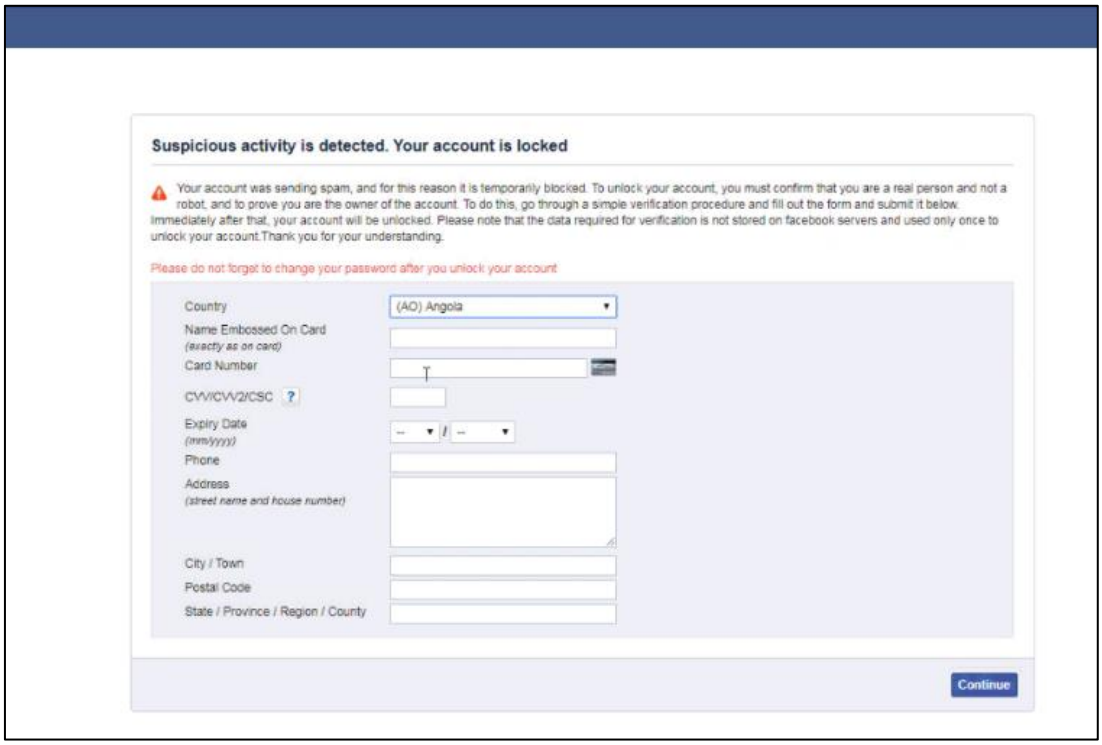
شکل شماره ۱۳ و شکل شماره ۱۴ آنچه کاربر بعد از ورود به وب‌سایت Facebook مشاهده می‌کند را نشان می‌دهد.



شکل شماره ۱۲



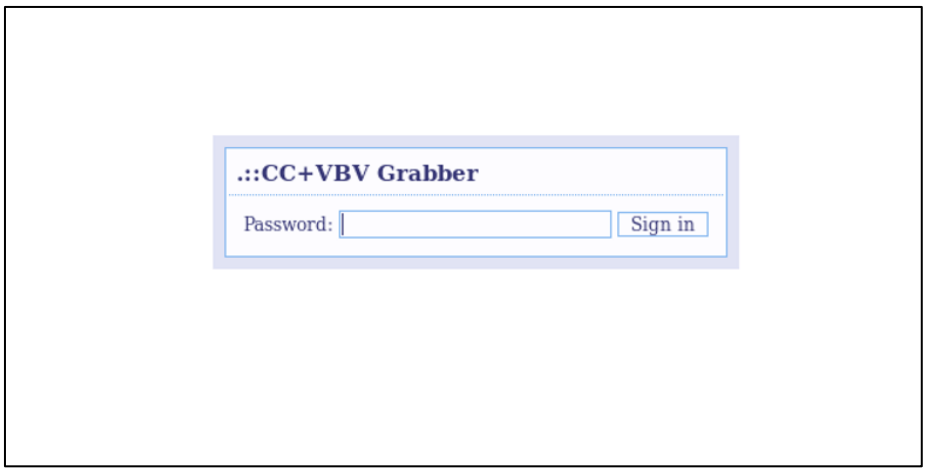
شکل شماره ۱۳



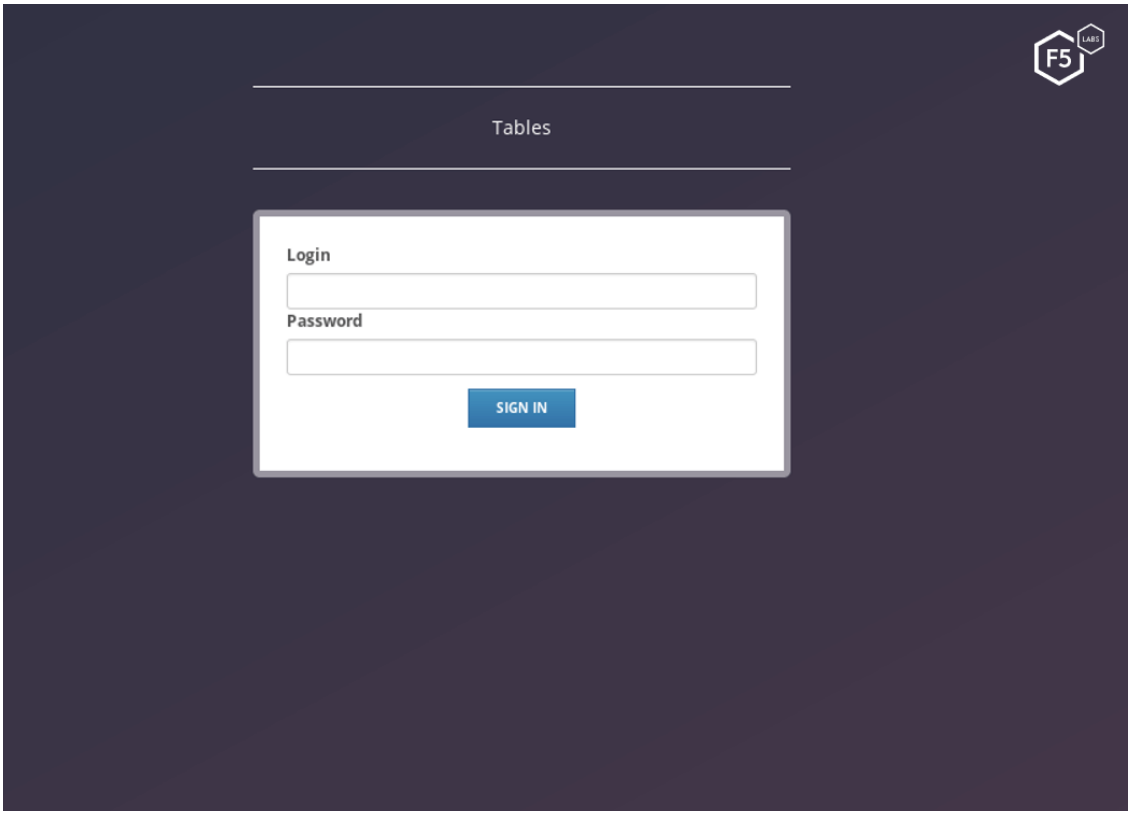
شکل شماره ۱۴

۳ زیرساخت حمله DanaBot

بعد از استفاده از روش‌های مطرح شده در بخش‌های قبل و یا دیگر ماژول‌های webinject، مهاجم می‌تواند اطلاعات حساس قربانی را از طریق مکانیسم CNC استخراج کند. مکانسیم VBV که از p.a.c.k.e.r استفاه می‌کند و جدول‌ها، هر کدام سرور CNC و پنل جداگانه‌ای دارند.

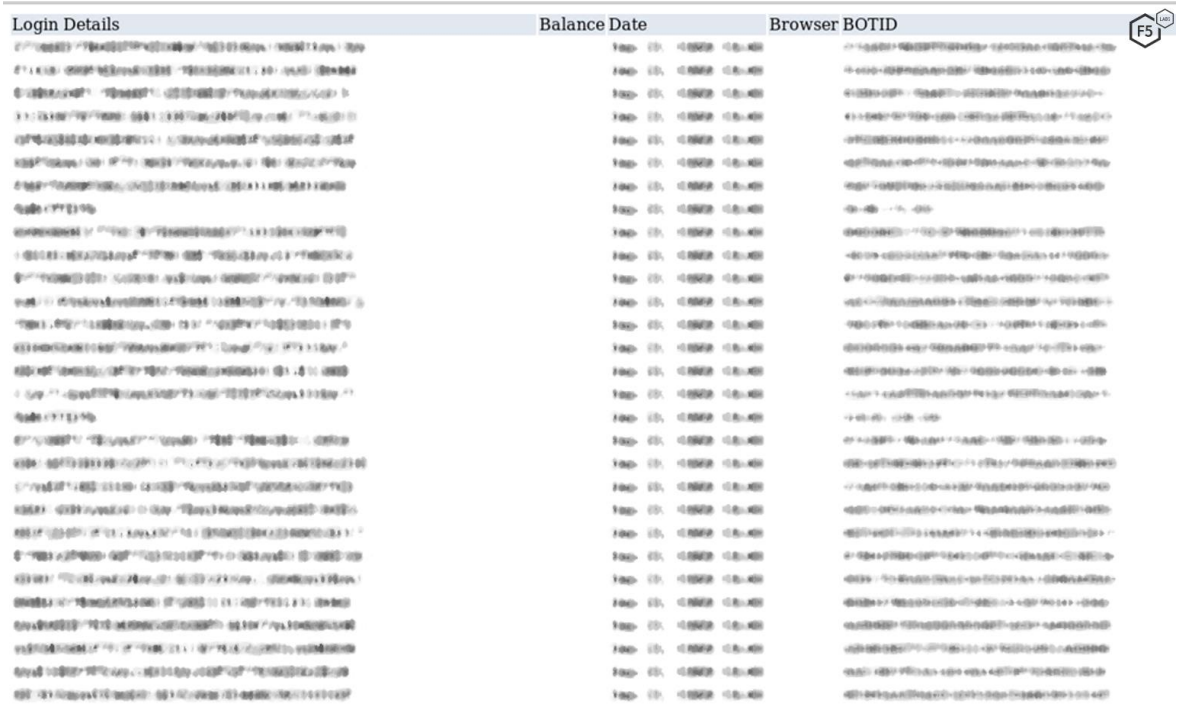


شکل شماره ۱۵:صفحه ورود به vbv grabber



شکل شماره ۱۶: صفحه ورود به جداول

محققان F5 توانستند به هر دو صفحات نشان داده شده در شکل شماره ۱۵و شکل شماره ۱۶ وارد شوند و همان داده‌هایی را مهاجمان DanaBot مشاهده کنند بدست آورند. با ورود به پنل کاربری مهاجم DanaBot اطلاعات واقعی قربانیان و BOTID مرورگر در دسترس است.



شکل شماره ۱۷: اطلاعات واقعی که از مرورگر قربانی به سرور DanaBot فرستاده شده است. (با توجه به حساسیت اطلاعات، تصویر مات شده است)

۴ نتیجه گیری

همانند باقی تروجان‌های بانکی، DanaBot نیز دائما در حال تغییر روش‌ها، اهداف و تاکتیک‌های خود برای پنهان ماندن از دید و ادامه دادن به کار خود می‌باشد. آنچه قابل توجه است تغییر هدف این تروجان از سرویس‌های موسسات مالی به سایت‌های ecommerce و streaming است.

احتمال ارتباط این بدافزار با تروجان‌های Gozi و Tinba با توجه به الگوهای مشترک injection استفاده شده در آنها قوت می‌گیرد با این حال اخیرا DanaBot یک بدافزار «Zeus مانند» نیز نامیده شده است. با توجه به پیشرفت و موفقیت روش‌های مورد استفاده این بدافزار پیش بینی می‌شود DanaBot یکی از بازیگران اصلی در حوزه تروجان‌های بانکی در سال ۲۰۲۰ باشد.

۵ منبع

[1] https://thehackernews.com/2020/01/hack-tiktok-account.html