
KNF
CSIRT
ANALIZA ZŁOŚLIWEJ APLIKACJI
MOBILNEJ IKO LOKATA

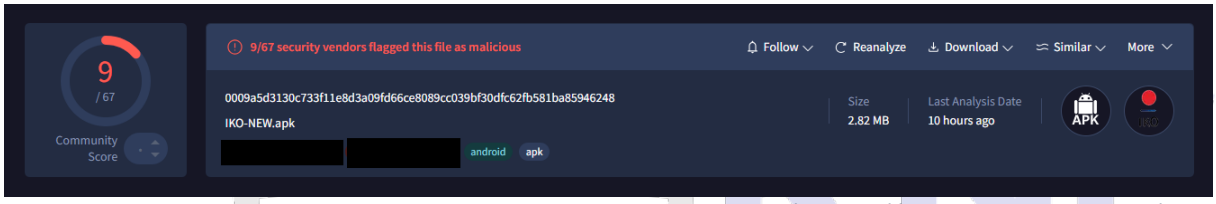
Analiza złośliwej aplikacji mobilnej IKO Lokata

Wprowadzenie

W ostatnim czasie zespół CSIRT KNF zaobserwował na portalu Facebook fałszywe reklamy, zachęcające do pobrania złośliwego oprogramowania, podszywającego się pod nieistniejącą aplikację „IKO Lokata". Kampania wycelowana była w użytkowników urządzeń mobilnych z systemem Android. Zachęcamy do zapoznania się z bieżącą analizą kampanii, a także potencjalnymi konsekwencjami zainstalowania złośliwej aplikacji na urządzeniu.

Próbka w serwisie VirusTotal

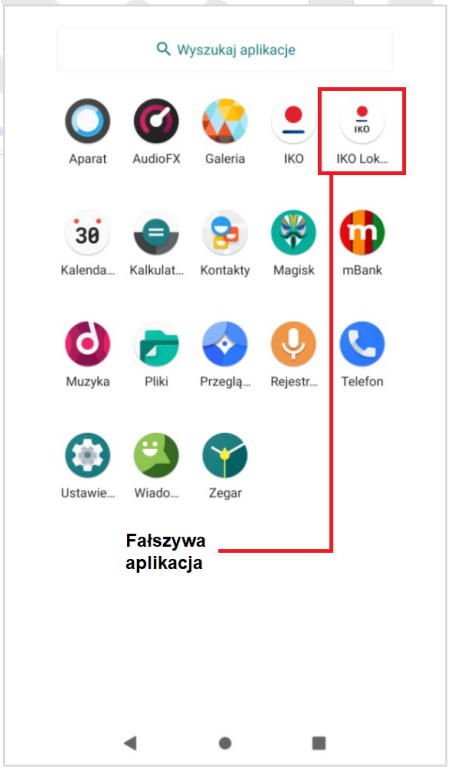
Punktem wyjścia do rozpoczęcia analizy, była obecna w serwisie VirusTotal, podejrzana próbka o nazwie **IKO-NEW.apk**, legitymująca się ikoną zawierającą elementy logotypu jednego z polskich banków.



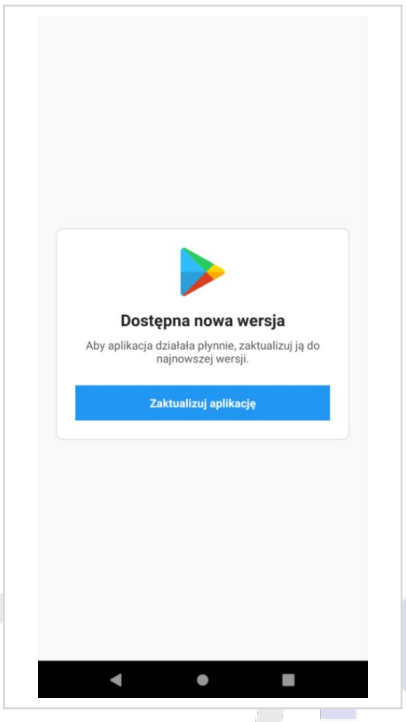
Podejrzany plik APK w serwisie VirusTotal

Pobranie i instalacja

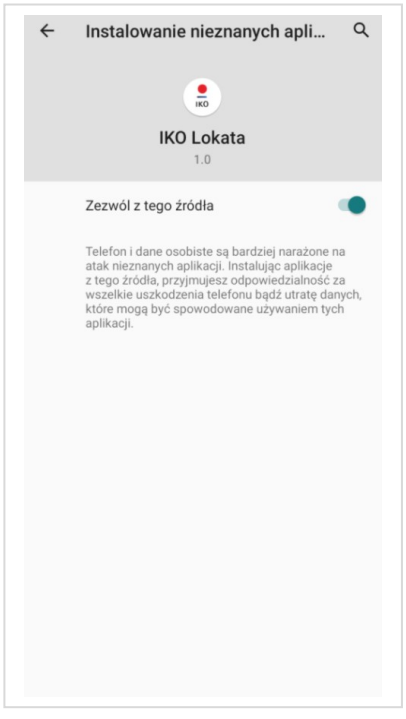
Po zainstalowaniu próbki na urządzeniu mobilnym widzimy, że zarówno zastosowany element wizualny w postaci ikony, jak i nazewnictwo, przypominają stylistykę aplikacji bankowej IKO.



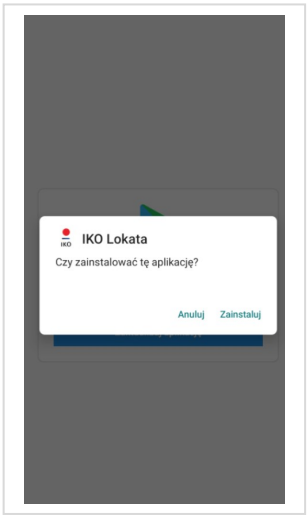
Uruchomienie aplikacji, skutkuje wyświetleniem okna informującego o dostępności nowej wersji produktu, zachęcającego do aktualizacji.



Kliknięcie w pole aktualizacji, przenosi nas do ekranu, w którym możemy zaakceptować instalację aplikacji z tzw. **niezaufanych źródeł**. Urządzenie ostrzega, że wyrażenie zgody naraża nas na potencjalne ryzyko, dając jednak prawo do kontynuowania, poprzez przesunięcie odpowiedniego suwaka.



Na potrzeby analizy wyrażamy zgodę na instalację aplikacji:

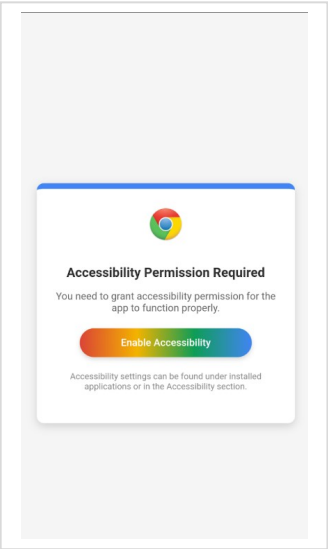


Kliknięcie przycisku **Zainstaluj** nie spowoduje pojawienia się dodatkowej ikony w menu aplikacji na urządzeniu. O tym, że dodatkowy pakiet znalazł się na naszym telefonie, możemy dowiedzieć się korzystając np. z interfejsu **ADB**. Odczytując listing z interesującego nas wyciągu aplikacji, możemy obok początkowo pobranego pakietu złośliwej próbki **purge.tremble**, zaobserwować pojawienie się dodatkowej paczki **unrelated.hamburger**.

Lista tzw. third-party packages przed i po wyrażeniu zgody na instalację

W tym miejscu warto zauważyć, że na potrzeby badania, analizę prowadzono częściowo na różnych urządzeniach. W badaniu wykorzystano urządzenia fizyczne z systemem Lineage 18.1-20220117-UNOFFICIAL-{zredagowano} oraz Android 11 z wyłączoną usługą Google Play Protect, która w przypadku działania, wykrywała złośliwą aktywność powiązaną z aplikacją IKO Lokata (w konsekwencji wiązało się to z podejmowaniem dodatkowych kroków w celu ominięcia ostrzeżeń).

Ukończenie procesu instalacji przenosi nas na ekran, w którym jesteśmy nakłaniani do przydzielenia tzw. **Uprawnień Dostępności** (ang. *Accessibility Services*). Funkcjonalność ta ma na celu m.in. pomoc w obsłudze urządzenia osobom z niepełnosprawnościami. Ze względu na fakt, że przydzielenie dostępu do interfejsu Dostępności może umożliwić aplikacji podejmowanie aktywności na urządzeniu w naszym imieniu, wspomniana funkcja bywa chętnie wykorzystywana w złośliwym oprogramowaniu przez atakujących.



Plansza informująca o konieczności przydzielenia złośliwej aplikacji uprawnień związanych z Dostępnością

Naszą uwagę przykuł tęczowy przycisk *Enable Accessibility,* przenoszący nas do kolejnego ekranu. Ten motyw graficzny miał okazję pojawiać się w opisywanej przez analityków **ThreatFabric** analizie złośliwego oprogramowania *Crocodilus*[1], co może wskazywać, że mamy do czynienia z wyżej wymienioną rodziną malware.

Kolejną rzeczą wzbudzającą zainteresowanie, była zaobserwowana na *proxy* komunikacja do serwera *wikimedia.org*. Na podstawie umieszczonego poniżej requestu wnioskujemy, że próbka pobiera ze wspomnianego serwera, widoczną na wcześniejszym ekranie ikonę przeglądarki Google Chrome.

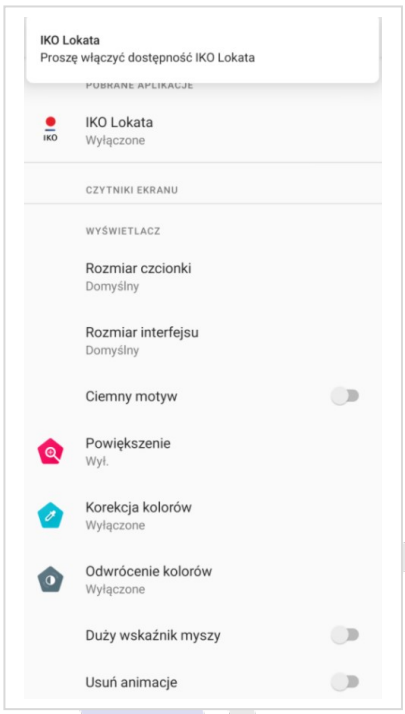
Request

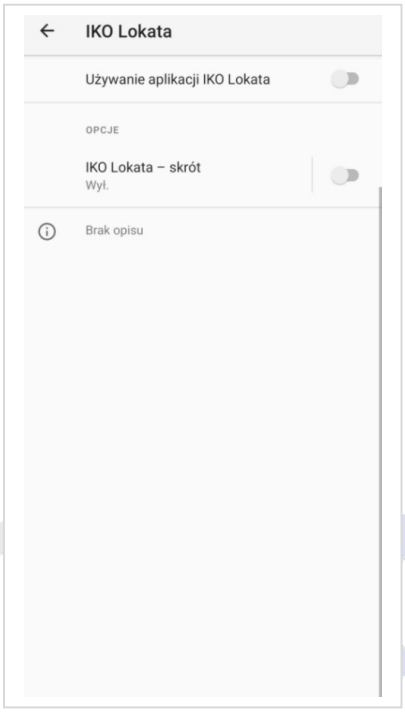
Pretty Raw Hex

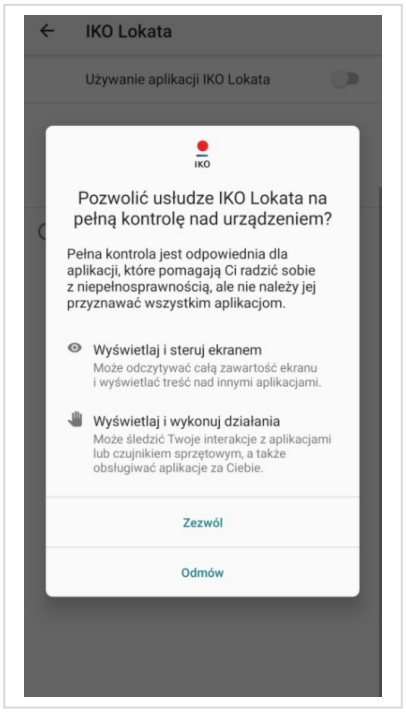
```
1 GET /wikipedia/commons/8/87/Google_Chrome_icon_%282011%29.png HTTP/2
2 Host: upload.wikimedia.org
3 User-Agent: [redacted]
4 Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
5 X-Requested-With: unrelated.hamburger
6 Sec-Fetch-Site: cross-site
7 Sec-Fetch-Mode: no-cors
8 Sec-Fetch-Dest: image
9 Accept-Encoding: gzip, deflate, br
10 Accept-Language: pl-PL,pl;q=0.9,en-US;q=0.8,en;q=0.7
```

[1] https://www.threatfabric.com/blogs/exposing-crocodilus-new-device-takeover-malware-targeting-android-devices

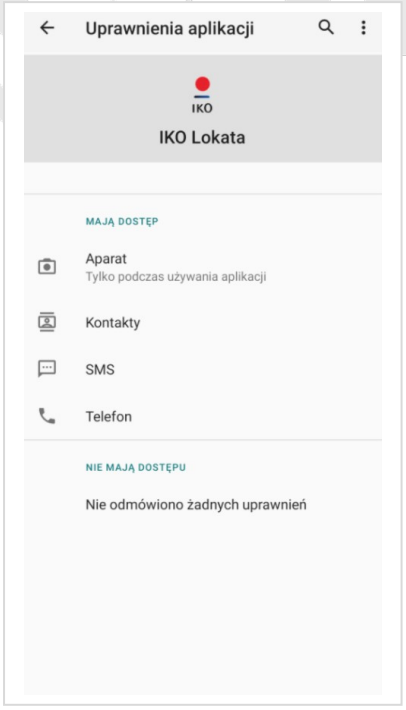
Kolejne kroki prowadzą nas do docelowego ekranu przydzielania uprawnień, związanych z **Dostępnością**. Jednocześnie urządzenie informuje nas o potencjalnych konsekwencjach wyrażenia zgody:







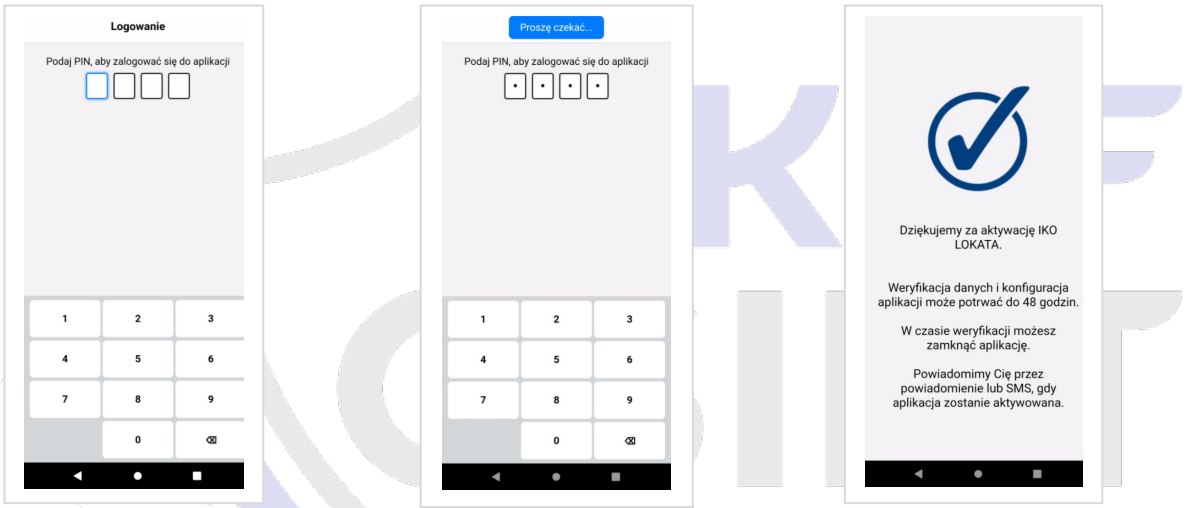
Akceptacja zgody i późniejsza weryfikacja dostępów wskazuje, że fałszywa aplikacja posiada uprawnienia typu Kontakty, Telefon czy SMS, których samodzielnie nie przydzielaliśmy:



Wyrażenie zgody na korzystanie z uprawnień *Accessibility* na drugim z testowanych urządzeń okazało się odrobinę bardziej problematyczne (w części przypadków aktywowała się ochrona przed kliknięciem w przycisk akceptacji, spowodowana przysłonięciem ekranu inną aplikacją), po chwili udało się jednak przydzielić dostęp do funkcji również w tym przypadku.

Zachowanie próbki na urządzeniu

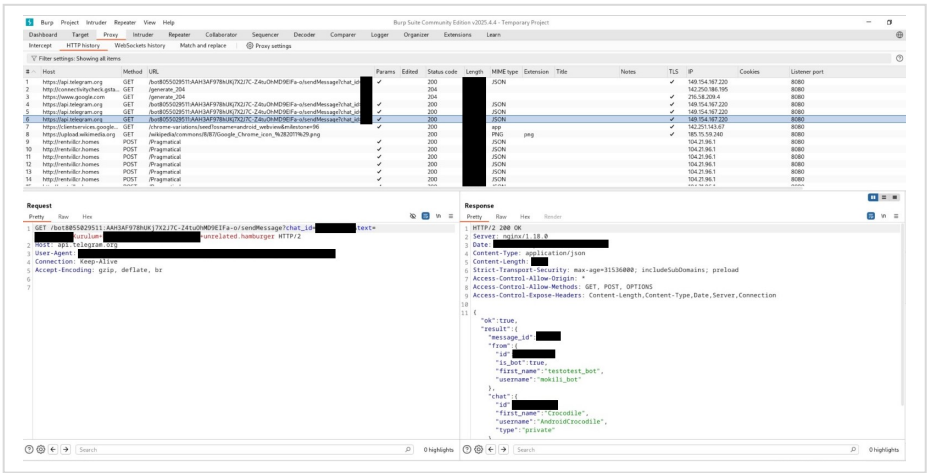
Po zakończonym procesie infekcji, podjęliśmy decyzję o dalszej obserwacji zachowania aplikacji w środowisku testowym. Zwróciliśmy uwagę, że w wybranych sytuacjach malware podejmował aktywność związaną z przykrywaniem okna aplikacji bankowej IKO i wyświetlaniem ekranu wnioskującego o wprowadzenie kodu PIN (wariant testowany na niezrootowanym urządzeniu z Androidem 11). Tego typu formularz wyświetlał się również po bezpośrednim uruchomieniu malware z menu aplikacji na urządzeniu. Co istotne, w tym drugim przypadku udało nam się dodatkowo zaobserwować sieciowy ruch wychodzący, zawierający wprowadzony PIN oraz ekran z „podziękowaniem" za aktywację aplikacji.



Komunikacja sieciowa

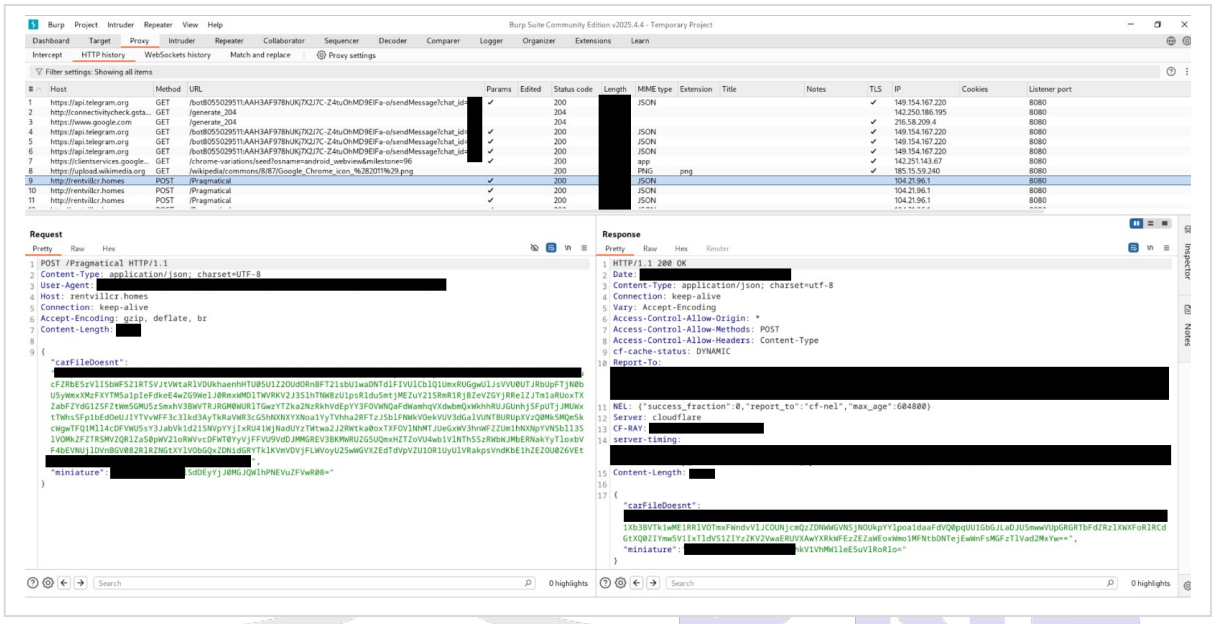
W trakcie analizy zaobserwowaliśmy komunikację sieciową, powiązaną m.in. z domenami:

- api.telegram[.]org



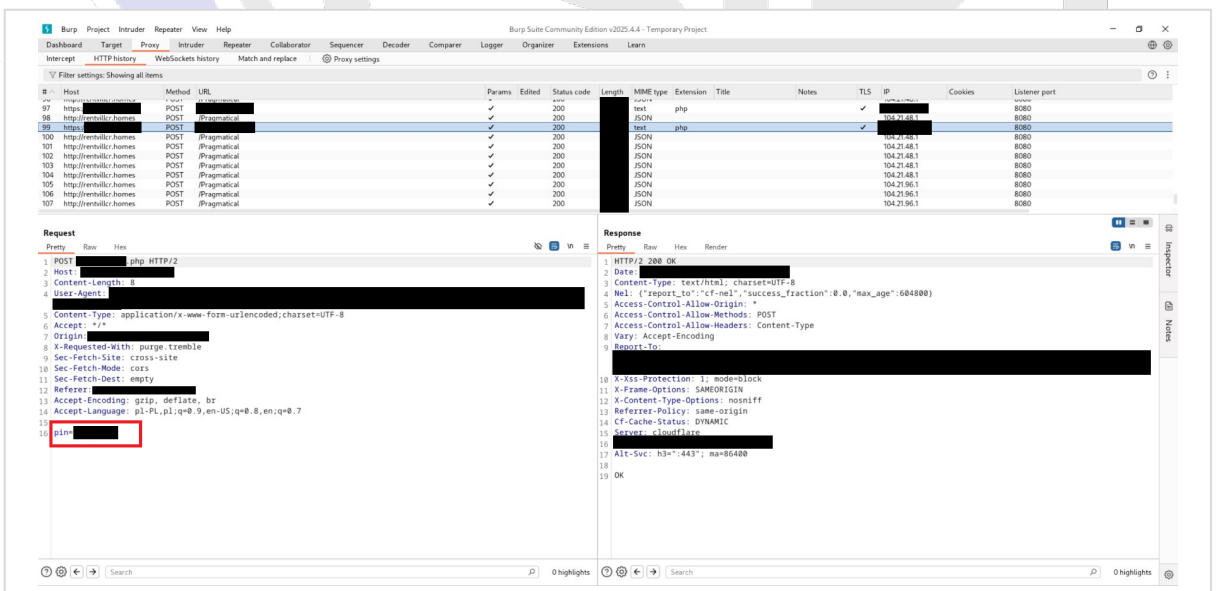
Obecność tagów Crocodile oraz AndroidCrocodile jest dla nas kolejną sugestią, że możemy mieć do czynienia z malware Crocodilus.

- rentvillcr[.]homes



*Endpoint **/Pragmatical**, tagi **carFileDoesnt** oraz **miniature**, również były obserwowane w przeszłości w kontekstach wiązanych z rodziną Crocodilus[2].*

- {domena zredagowana}



Adres, na który kierowany był zaobserwowany request z kodem PIN.

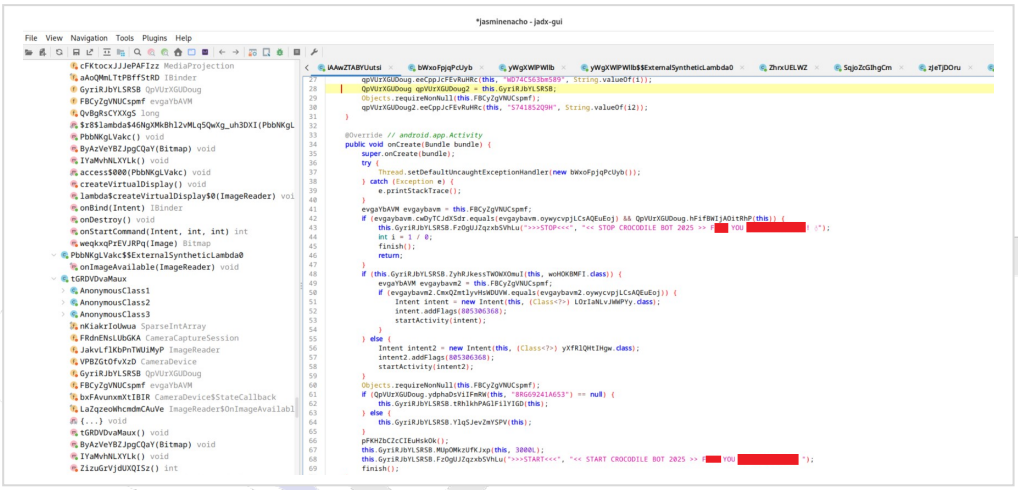
[2] https://x.com/naumovax/status/1906727042353107402

Ciekawostki z analizy statycznej

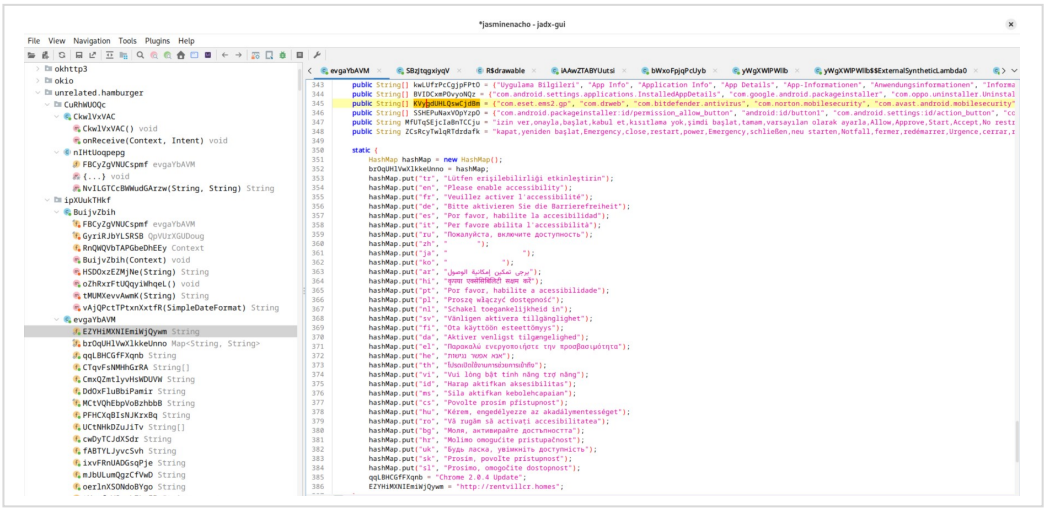
Podczas śledzenia mapy pamięci wybranych procesów działających na urządzeniu, nasze zainteresowanie wzbudził usunięty plik w formacie *Dalvik Executable*:

```
9560f000-9760f000 r--s 00000000 00:06 22190                              /dev/ashmem/jit-zygote-cache (deleted)
9760f000-9960f000 r-xs 02000000 00:06 22190                              /dev/ashmem/jit-zygote-cache (deleted)
9960f000-9b60f000 r--s 00000000 00:06 95153                              /memfd:jit-cache (deleted)
9b60f000-9d60f000 r-xs 02000000 00:06 95153                              /memfd:jit-cache (deleted)
7136f47000-7136f67000 rw-s 00000000 00:06 97147                          /dev/ashmem/shared_memory/FDF793F4BC996CC2699D31ED90C99EB9 (deleted)
7139435000-7139455000 rw-s 00000000 00:06 97146                          /dev/ashmem/shared_memory/FC1DA83DF54F0478F00658632A0C1D93 (deleted)
713fbf9000-713fcdb000 r--p 00000000 103:10 3015015                       /data/data/unrelated.hamburger/code_cache/jasminenacho.dex (deleted)
7142387000-714238f000 rw-s 00000000 00:06 95200                          /dev/ashmem/shared_memory/C25018F3D01E87F1FFB2823874AD7F3B (deleted)
71447cb000-71467cb000 rw-s 00000000 00:06 95153                          /memfd:jit-cache (deleted)
71467cb000-71487cb000 r--s 02000000 00:06 95153                          /memfd:jit-cache (deleted)
73e4cfc000-73e4cfd000 rw-s 00000000 00:06 97145                          /dev/ashmem/shared_memory/CA37A33E1746A5865FF4605A9601E272 (deleted)
73e52a2000-73e52a3000 rw-s 00000000 00:06 97351                          /dev/ashmem/GFXStats-13367 (deleted)
73e53f6000-73e53f7000 rw-s 00000000 00:06 97145                          /dev/ashmem/shared_memory/CA37A33E1746A5865FF4605A9601E272 (deleted)
73e5470000-73e5471000 r--s 00000000 00:06 27085                          /dev/ashmem/9cd34a41-464e-46ac-b125-3474f45017d2 (deleted)
73e57ef000-73e57f0000 r--s 00000000 00:06 27085                          /dev/ashmem/9cd34a41-464e-46ac-b125-3474f45017d2 (deleted)
73e77de000-73e77df000 r--s 00000000 00:06 27085                          /dev/ashmem/9cd34a41-464e-46ac-b125-3474f45017d2 (deleted)
```

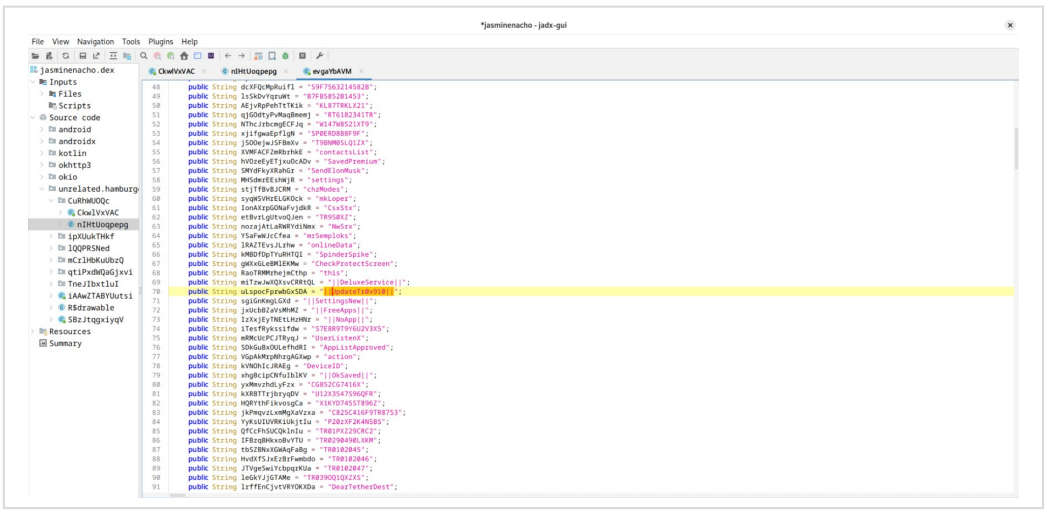
Z uwagi na fakt, że plik mógł zawierać przydatne informacje z punktu widzenia dalszej analizy, postanowiliśmy go pozyskać i poddać procesowi dekompilacji. Poniżej publikujemy ciekawsze fragmenty, celem uzupełnienia opisu:



*Zdekompilowany fragment złośliwej próbki – zapisy **START/STOP CRODODILE BOT 2025** stanowią kolejną podpowiedź, że możemy mieć do czynienia z malware Crocodilus. W kodzie znalazły się również obraźliwe komentarze, adresowane jak mniemamy, w kierunku jednego z analityków malware – zostały one zredagowane kolorem czerwonym.*



*Zapisy wskazujące m.in. na zaobserwowaną wcześniej domenę, jak również nazwy pakietów, sugerujące **rozwiązania antywirusowe** (hipotezą ich umieszczenia w tym miejscu, może być weryfikacja urządzenia pod kątem obecności ww. aplikacji przez malware).*

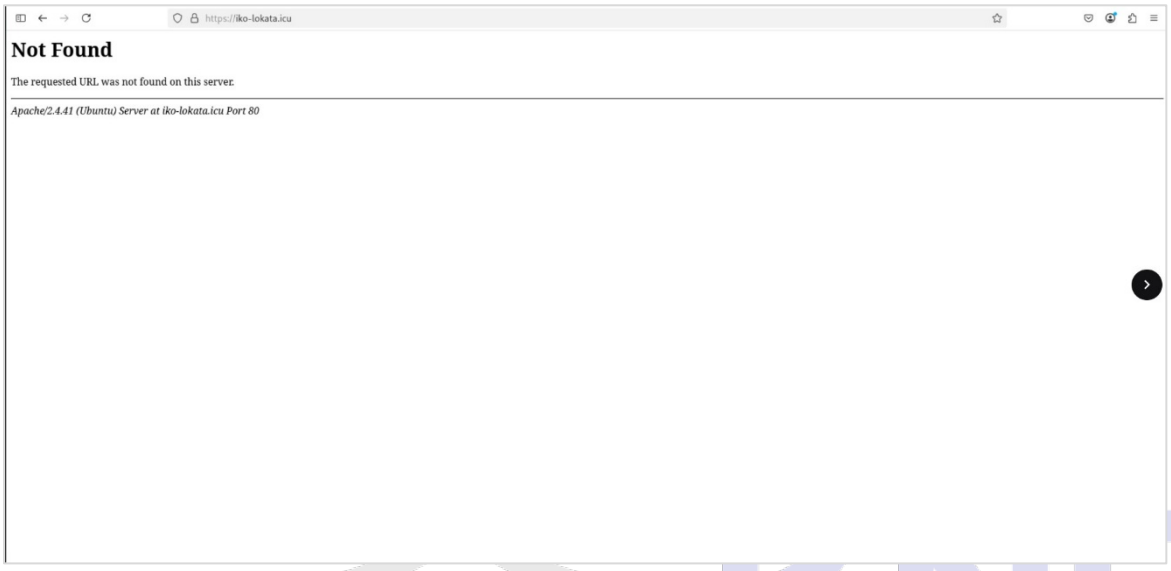


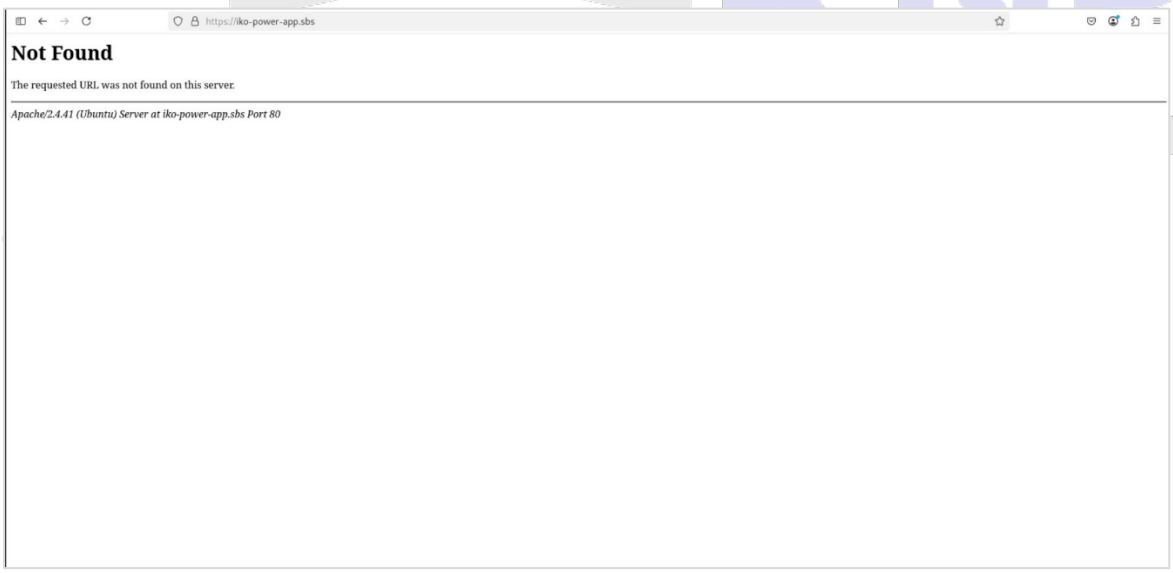
Odnalezione łańcuchy znakowe. Część z nich, m.in. ***||UpdateTr0x910||****,* ***mkLoper****,* ***CsxStx****,* ***||SettingsNew||****,* ***||FreeApps||****,* ***SpinderSpike****, czy* ***MrSemploks*** *– pojawia się we wspomnianym wcześniej raporcie ThreatFabric w kontekście komend obsługiwanych przez malware Crocodilus.*

Sposób dystrybucji

Pobrana przez nas próbka złośliwego oprogramowania pochodziła z serwisu VirusTotal. Równolegle, mieliśmy okazję odnaleźć obecne na platformie Facebook złośliwe reklamy, wpisujące się w stylistykę analizowanej aplikacji **IKO Lokata**.

Odwiedzone adresy, do których prowadziły zaobserwowane reklamy, w momencie analizy nie posiadały skonfigurowanych witryn:





Podsumowanie i rekomendacje

Analizowana próbka malware podszywa się pod rzekomo oficjalną aplikację „IKO lokata", która w rzeczywistości **nie istnieje**. W opisanym przypadku, analizowane artefakty wskazują, że możemy mieć do czynienia ze złośliwym oprogramowaniem z rodziny Crocodilus. Pomimo, iż próbka została przez nas pobrana z zewnętrznego serwisu analitycznego, obecność oszukańczych reklam w mediach społecznościowych **nie wyklucza pojawienia się jej w przyszłości na fałszywych stronach internetowych**. Zachęcamy do zachowania ostrożności oraz instalowania aplikacji, pochodzących wyłącznie z zaufanych źródeł.

IoC

Filename: IKO-NEW.apk
Package name: purge.tremble
MD5: 689579531a417b84ddbceb17c75d3c39
SHA-1: afc8dee366c4a0ac06bbda3253951f11440e1572
SHA-256: 0009a5d3130c733f11e8d3a09fd66ce8089cc039bf30dfc62fb581ba85946248
Cert thumbprint: c45471a904ea66f29a991e08e6be296a63a2a1cd

Filename: fstream-6.apk
Package name: unrelated.hamburger
MD5: e7551da0d6e05cce11d4bf3ae016bb15
SHA-1: 9e14d550c6b2ad3f1ff0a46c86b66ab1f262e2fe
SHA-256: 801f956057aa608c7393b2fcca9b731c610c6bd5e14f4f3c1d64e8f211828319
Cert thumbprint: 9ef0a6f1d70519270d50f24aeb7d3e897085fa80

Filename: jasminenacho.dex
MD5: efeeb020bb4f0bb13c7f2971a6a658a3
SHA-1: 090a2157cafff824df2bd5fa5848125579117aa7
SHA-256: 723b11db3994b335a19f594381396773c4064469f0293a73fc6f4f2d6c6fef6b

- hxxps://www.facebook[.]com/profile.php?id=61576199655216
- hxxps://www.facebook[.]com/profile.php?id=61576237809007
- hxxps://iko-power-app[.]sbs
- hxxps://iko-lokata[.]icu
- rentvillcr[.]homes
- api.telegram[.]org
- /Pragmatical