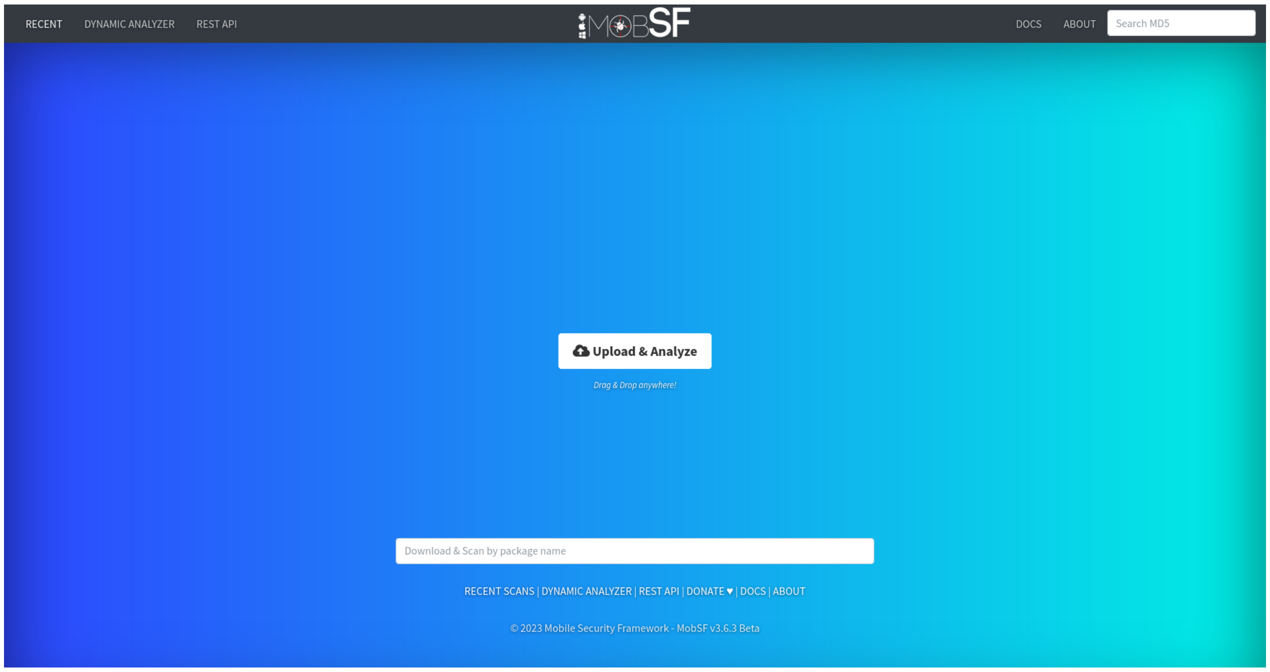
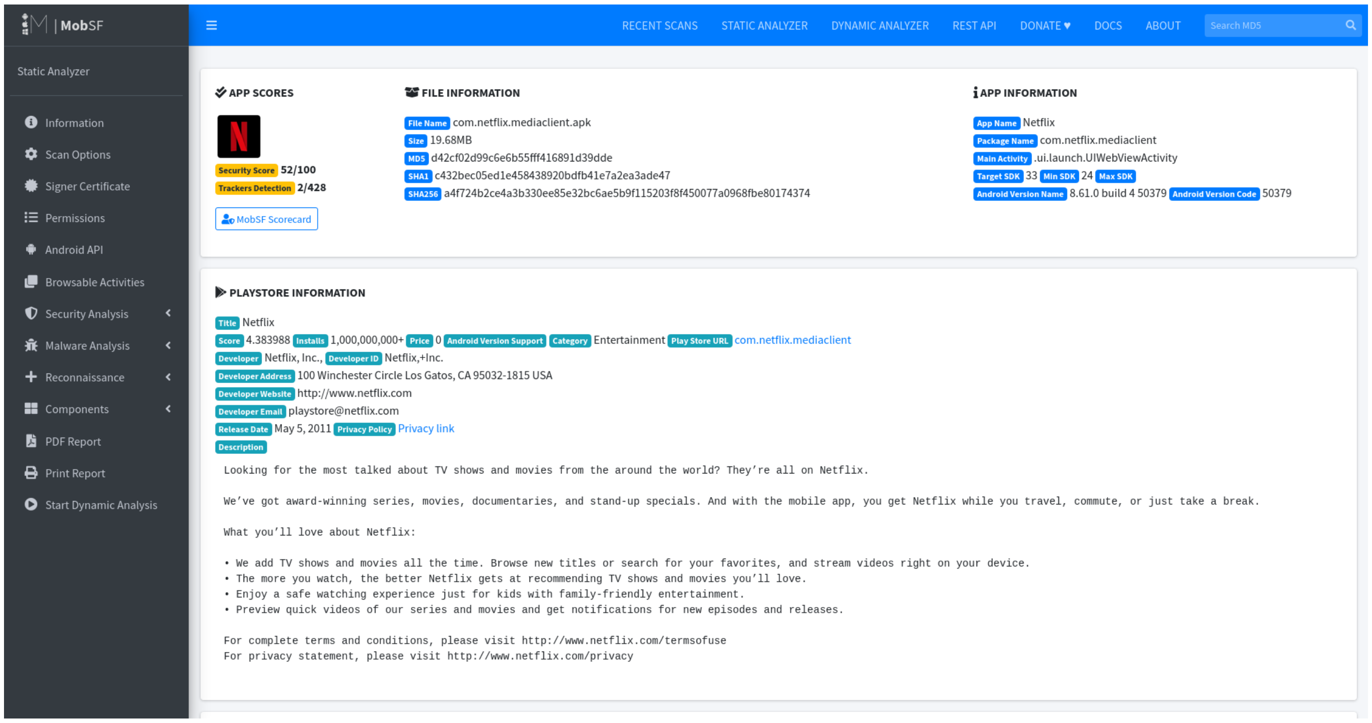
MOBSF DEMO

Installation: https://allabouttesting.org/quick-tutorial-mobsf-installation-on-linux-windows/

After the installation, you can launch the application and see the following interface.


RECENT
DYNAMIC ANALYZER
REST API
MobSF
DOCS
ABOUT
Search MD5
Upload & Analyze
Drag & Drop anywhere!
Download & Scan by package name
RECENT SCANS | DYNAMIC ANALYZER | REST API | DONATE ♥ | DOCS | ABOUT
© 2023 Mobile Security Framework - MobSF v3.6.3 Beta

As you can see, there is a button we can use to upload an apk to analyze. We can also specify the name of the package we want to analyze, and MobSF will download it for us from apktada.com.
As a sample application, we will analyze the netflix app. After the analysis ends, we can start inspecting the report.
First, we can see the general information of the application: the file name, various hashes, as well as its main activity and the versions of the android SDK it uses. On the left side of the page, there is a menu which lists the different sections of the report: first, there are some sections with general information, then there are specific sections related to the app security and the malware analysis.



APP SCORES

Security Score 52/100
Trackers Detection 2/428

FILE INFORMATION

File Name com.netflix.mediaclient.apk
Size 19.68MB
MD5 d42cf02d99c6e6b55fff416891d39dde
SHA1 c432bec05ed1e458438920bdfb41e7a2ea3ade47
SHA256 a4f724b2ce4a3b330ee85e32bc6ae5b9f115203f8f450077a0968fbe80174374

APP INFORMATION

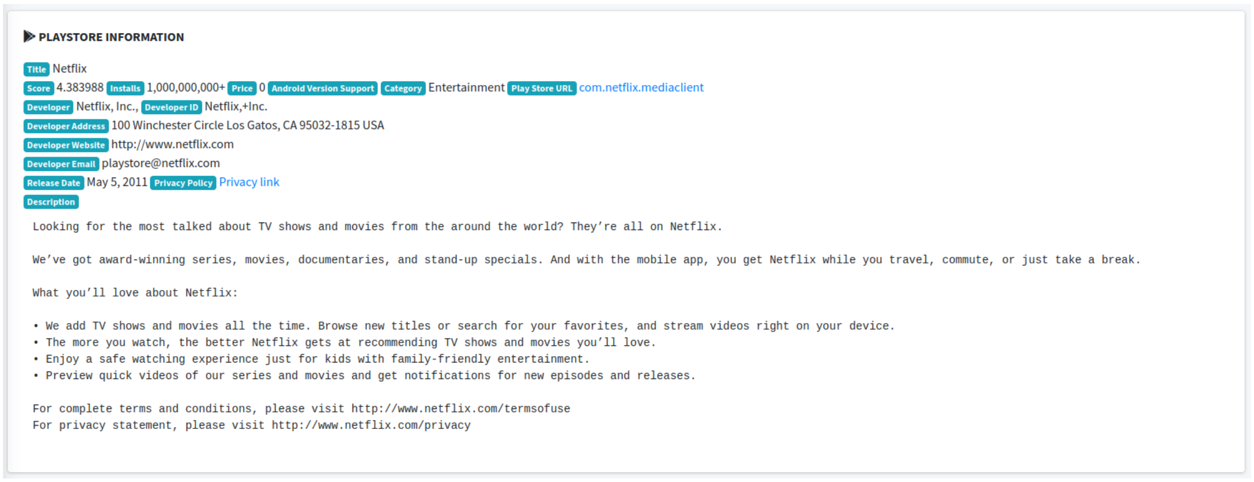
App Name Netflix
Package Name com.netflix.mediaclient
Main Activity .ui.launch.UIWebViewActivity
Target SDK 33 Min SDK 24 Max SDK
Android Version Name 8.61.0 build 4 50379 Android Version Code 50379

PLAYSTORE INFORMATION

Title Netflix
Score 4.383988 Installs 1,000,000,000+ Price 0 Android Version Support Category Entertainment Play Store URL com.netflix.mediaclient
Developer Netflix, Inc., Developer ID Netflix,+Inc.
Developer Address 100 Winchester Circle Los Gatos, CA 95032-1815 USA
Developer Website http://www.netflix.com
Developer Email playstore@netflix.com
Release Date May 5, 2011 Privacy Policy Privacy link
Description

```
Looking for the most talked about TV shows and movies from the around the world? They’re all on Netflix.

We’ve got award-winning series, movies, documentaries, and stand-up specials. And with the mobile app, you get Netflix while you travel, commute, or just take a break.

What you’ll love about Netflix:

• We add TV shows and movies all the time. Browse new titles or search for your favorites, and stream videos right on your device.
• The more you watch, the better Netflix gets at recommending TV shows and movies you’ll love.
• Enjoy a safe watching experience just for kids with family-friendly entertainment.
• Preview quick videos of our series and movies and get notifications for new episodes and releases.

For complete terms and conditions, please visit http://www.netflix.com/termsofuse
For privacy statement, please visit http://www.netflix.com/privacy
```

Moving on, we can find the Play Store information section. This section contains all the information MobSF fetched from the Play Store, such as the number of downloads, number of installs, developer information, and so on.


PLAYSTORE INFORMATION

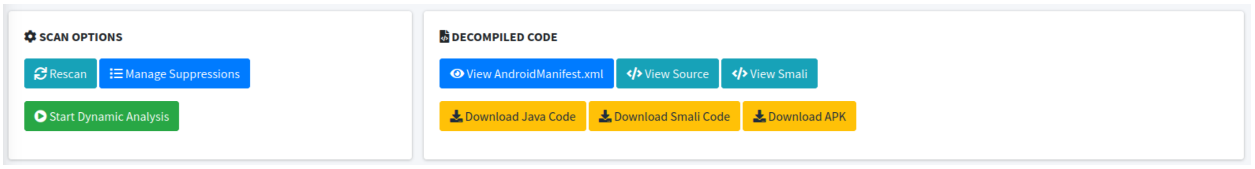
Title Netflix
Score 4.383988 Installs 1,000,000,000+ Price 0 Android Version Support Category Entertainment Play Store URL com.netflix.mediaclient
Developer Netflix, Inc., Developer ID Netflix,+Inc.
Developer Address 100 Winchester Circle Los Gatos, CA 95032-1815 USA
Developer Website http://www.netflix.com
Developer Email playstore@netflix.com
Release Date May 5, 2011 Privacy Policy Privacy link
Description

```
Looking for the most talked about TV shows and movies from the around the world? They’re all on Netflix.

We’ve got award-winning series, movies, documentaries, and stand-up specials. And with the mobile app, you get Netflix while you travel, commute, or just take a break.

What you’ll love about Netflix:

• We add TV shows and movies all the time. Browse new titles or search for your favorites, and stream videos right on your device.
• The more you watch, the better Netflix gets at recommending TV shows and movies you’ll love.
• Enjoy a safe watching experience just for kids with family-friendly entertainment.
• Preview quick videos of our series and movies and get notifications for new episodes and releases.

For complete terms and conditions, please visit http://www.netflix.com/termsofuse
For privacy statement, please visit http://www.netflix.com/privacy
```

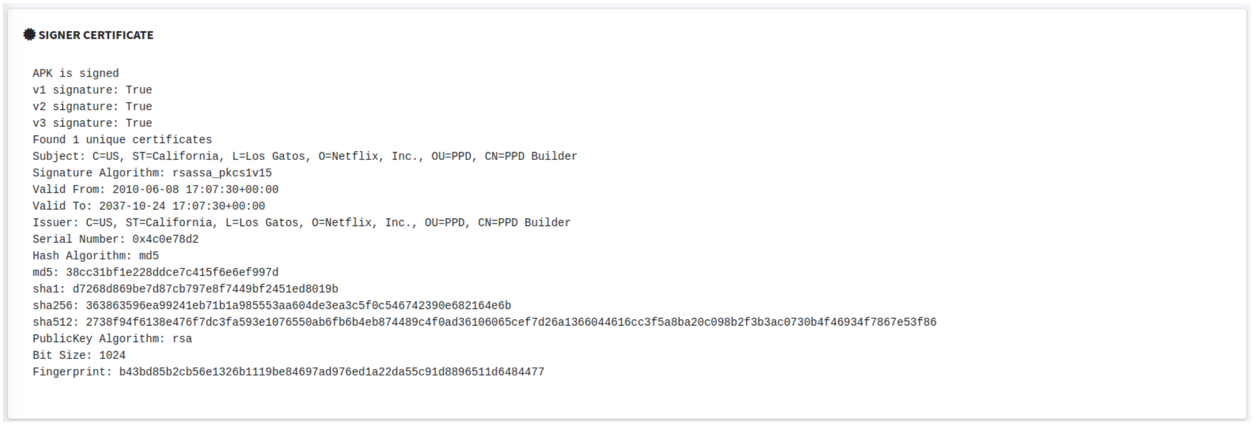
The next section is a summary about the various components of the app: it reports the number of components for each type (Activity, BroadcastReceiver, Service and Provider), as well as the number of exported components. This gives us an idea about the entry points we could use to find vulnerabilities in the app.

Next, we can see the decompiled code section. This section provides us with links we can use to access the files that were decompiled from the app, which we can use for manual analysis.

For example, clicking on the View AndroidManifest.xml button we can have a look at the extracted manifest of the application.

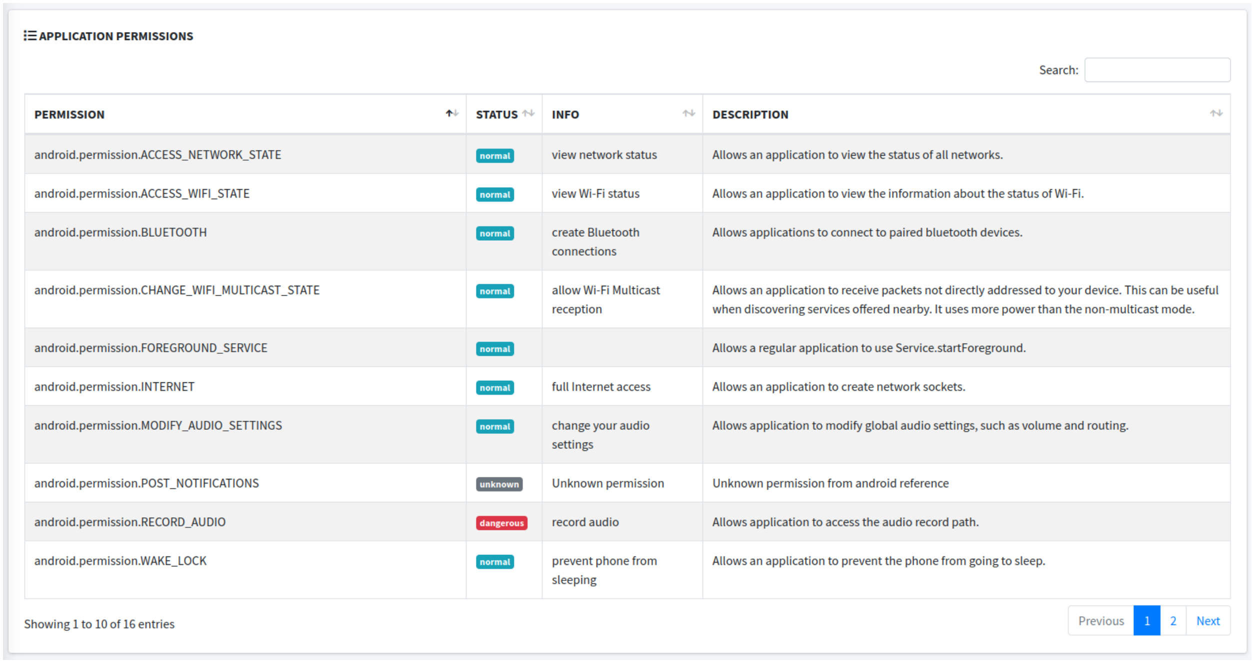


In the signer certificate section, MobSF reports all the information contained in the app certificate. This information can be useful to detect potential repackaged APK files, since we can compare the info reported on the apk with the developer info. In the security analysis section, MobSF performs some additional analysis on the app certificate, which we will see in a bit.


SIGNER CERTIFICATE

```
APK is signed
v1 signature: True
v2 signature: True
v3 signature: True
Found 1 unique certificates
Subject: C=US, ST=California, L=Los Gatos, O=Netflix, Inc., OU=PPD, CN=PPD Builder
Signature Algorithm: rsassa_pkcs1v15
Valid From: 2010-06-08 17:07:30+00:00
Valid To: 2037-10-24 17:07:30+00:00
Issuer: C=US, ST=California, L=Los Gatos, O=Netflix, Inc., OU=PPD, CN=PPD Builder
Serial Number: 0x4c0e78d2
Hash Algorithm: md5
md5: 38cc31bf1e228ddce7c415f6e6ef997d
sha1: d7268d869be7d87cb797e8f7449bf2451ed8019b
sha256: 363863596ea99241eb71b1a985553aa604de3ea3c5f0c546742390e682164e6b
sha512: 2738f94f6138e476f7dc3fa593e1076550ab6fb6b4eb874489c4f0ad36106065cef7d26a1366044616cc3f5a8ba20c098b2f3b3ac0730b4f46934f7867e53f86
PublicKey Algorithm: rsa
Bit Size: 1024
Fingerprint: b43bd85b2cb56e1326b1119be84697ad976ed1a22da55c91d8896511d6484477
```

The application permissions section lists all the permissions declared by the application. We can also sort the permissions by protection level (e.g. normal, dangerous), so that we see all the dangerous permissions used by the app. In this case, we can see that the app uses the RECORD_AUDIO and the WRITE_EXTERNAL_STORAGE dangerous permissions. The information in this section is useful to understand if an application declares some unexpected permissions: if for example a notes app declares the FINE_LOCATION permission, it may be an indication of some unwanted data collection.



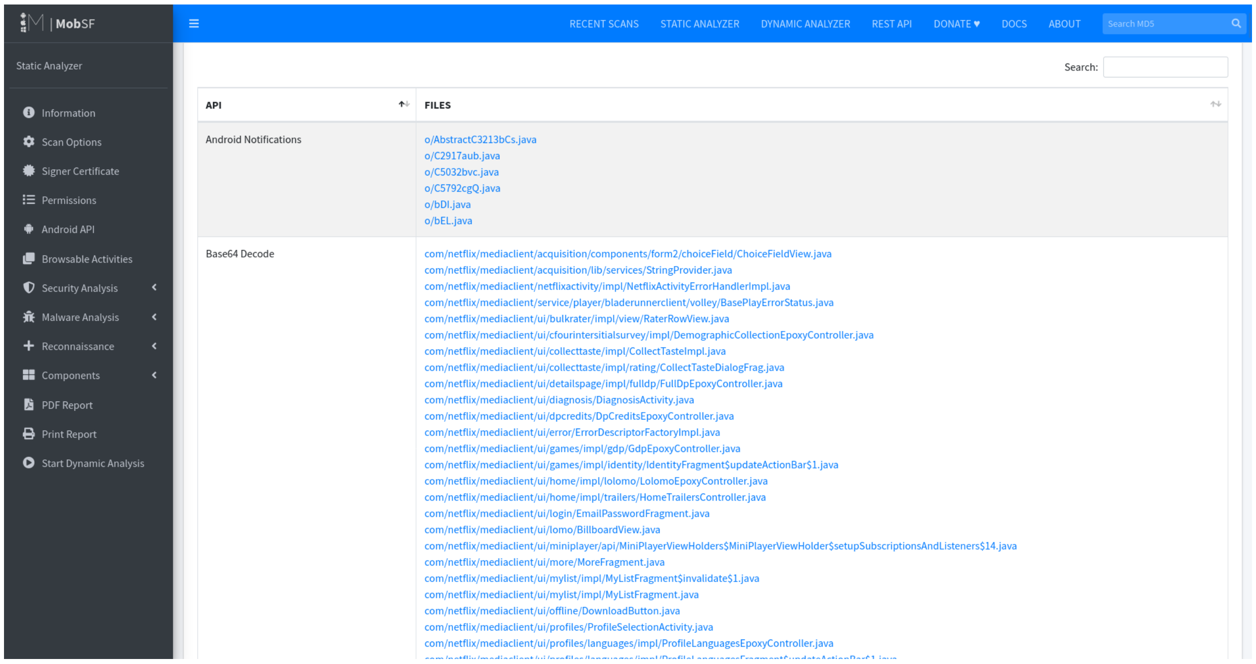
APPLICATION PERMISSIONS

Search:

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.ACCESS_NETWORK_STATE	normal	view network status	Allows an application to view the status of all networks.
android.permission.ACCESS_WIFI_STATE	normal	view Wi-Fi status	Allows an application to view the information about the status of Wi-Fi.
android.permission.BLUETOOTH	normal	create Bluetooth connections	Allows applications to connect to paired bluetooth devices.
android.permission.CHANGE_WIFI_MULTICAST_STATE	normal	allow Wi-Fi Multicast reception	Allows an application to receive packets not directly addressed to your device. This can be useful when discovering services offered nearby. It uses more power than the non-multicast mode.
android.permission.FOREGROUND_SERVICE	normal		Allows a regular application to use Service.startForeground.
android.permission.INTERNET	normal	full Internet access	Allows an application to create network sockets.
android.permission.MODIFY_AUDIO_SETTINGS	normal	change your audio settings	Allows application to modify global audio settings, such as volume and routing.
android.permission.POST_NOTIFICATIONS	unknown	Unknown permission	Unknown permission from android reference
android.permission.RECORD_AUDIO	dangerous	record audio	Allows application to access the audio record path.
android.permission.WAKE_LOCK	normal	prevent phone from sleeping	Allows an application to prevent the phone from going to sleep.

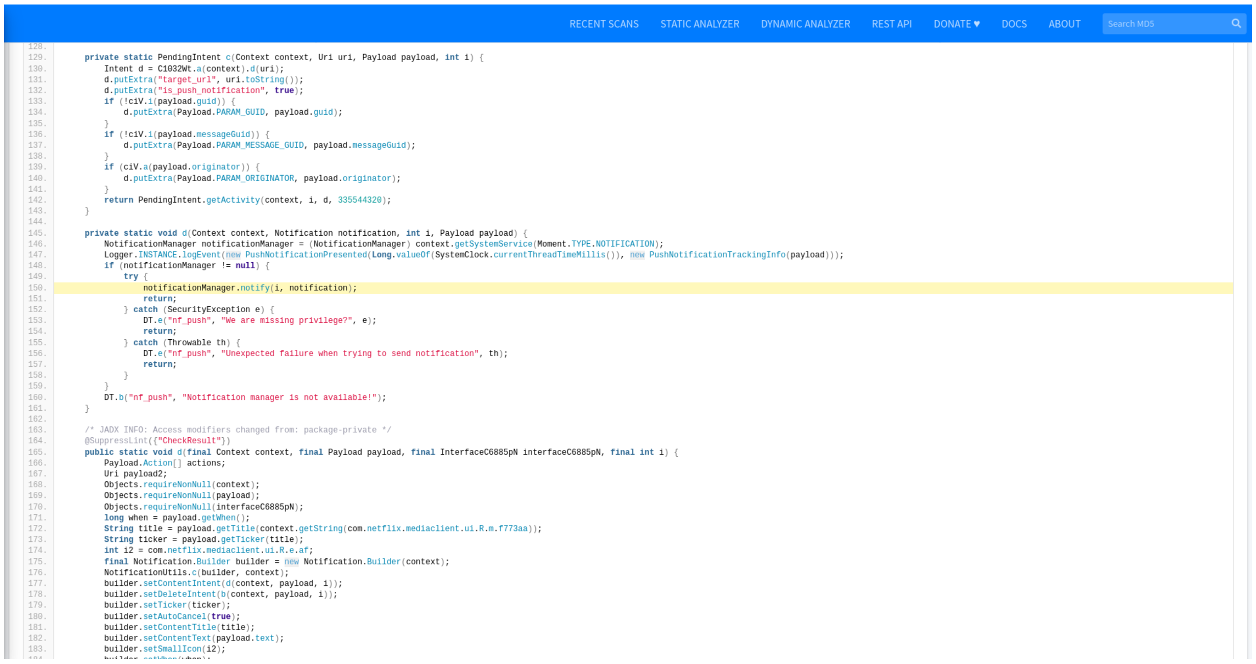
Showing 1 to 10 of 16 entries

The next section is the Android API section, which lists what kind of APIs are used by the app. For example, in this case, we see that the Netflix app uses some notification API. On the right we can also see the list of files where such operations were detected, which we can click to manually inspect the affected code.



Search:

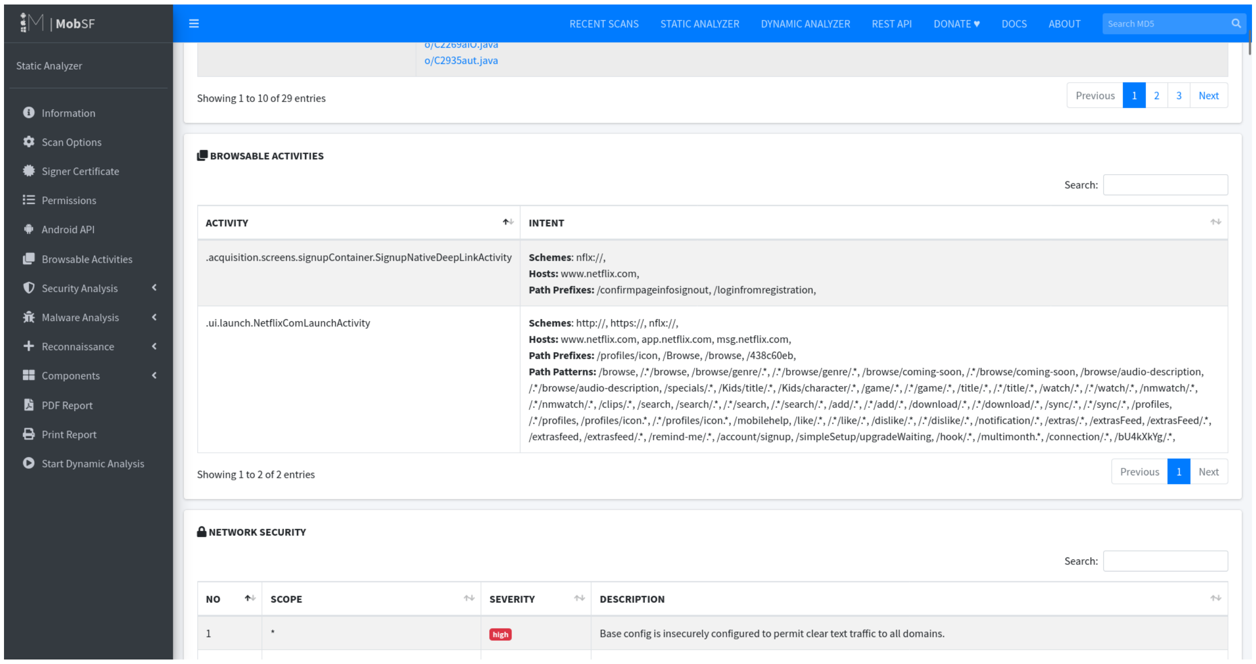
API	FILES
Android Notifications	o/AbstractC3213bCs.java o/C2917aub.java o/C5032bvc.java o/C5792cgQ.java o/bDI.java o/bEL.java
Base64 Decode	com/netflix/mediaclient/acquisition/components/form2/choiceField/ChoiceFieldView.java com/netflix/mediaclient/acquisition/lib/services/StringProvider.java com/netflix/mediaclient/netflixactivity/impl/NetflixActivityErrorHandlerImpl.java com/netflix/mediaclient/service/player/bladerunnerclient/volley/BasePlayErrorStatus.java com/netflix/mediaclient/ui/bulkrater/impl/view/RaterRowView.java com/netflix/mediaclient/ui/cfourintersitialsurvey/impl/DemographicCollectionEpoxyController.java com/netflix/mediaclient/ui/collecttaste/impl/CollectTasteImpl.java com/netflix/mediaclient/ui/collecttaste/impl/rating/CollectTasteDialogFrag.java com/netflix/mediaclient/ui/detailspage/impl/fulldp/FullDpEpoxyController.java com/netflix/mediaclient/ui/diagnosis/DiagnosisActivity.java com/netflix/mediaclient/ui/dpcredits/DpCreditsEpoxyController.java com/netflix/mediaclient/ui/error/ErrorDescriptorFactoryImpl.java com/netflix/mediaclient/ui/games/impl/gdp/GdpEpoxyController.java com/netflix/mediaclient/ui/games/impl/identity/IdentityFragment$updateActionBar$1.java com/netflix/mediaclient/ui/home/impl/lolomo/LolomoEpoxyController.java com/netflix/mediaclient/ui/home/impl/trailers/HomeTrailersController.java com/netflix/mediaclient/ui/login/EmailPasswordFragment.java com/netflix/mediaclient/ui/lomo/BillboardView.java com/netflix/mediaclient/ui/miniplayer/api/MiniPlayerViewHolders$MiniPlayerViewHolder$setupSubscriptionsAndListeners$14.java com/netflix/mediaclient/ui/more/MoreFragment.java com/netflix/mediaclient/ui/mylist/impl/MyListFragment$invalidate$1.java com/netflix/mediaclient/ui/mylist/impl/MyListFragment.java com/netflix/mediaclient/ui/offline/DownloadButton.java com/netflix/mediaclient/ui/profiles/ProfileSelectionActivity.java com/netflix/mediaclient/ui/profiles/languages/impl/ProfileLanguagesEpoxyController.java



```java

    private static PendingIntent c(Context context, Uri uri, Payload payload, int i) {
        Intent d = C1032Wt.a(context).d(uri);
        d.putExtra("target_url", uri.toString());
        d.putExtra("is_push_notification", true);
        if (!ciV.i(payload.guid)) {
            d.putExtra(Payload.PARAM_GUID, payload.guid);
        }
        if (!ciV.i(payload.messageGuid)) {
            d.putExtra(Payload.PARAM_MESSAGE_GUID, payload.messageGuid);
        }
        if (ciV.a(payload.originator)) {
            d.putExtra(Payload.PARAM_ORIGINATOR, payload.originator);
        }
        return PendingIntent.getActivity(context, i, d, 335544320);
    }

    private static void d(Context context, Notification notification, int i, Payload payload) {
        NotificationManager notificationManager = (NotificationManager) context.getSystemService(Moment.TYPE.NOTIFICATION);
        Logger.INSTANCE.logEvent(new PushNotificationPresented(Long.valueOf(SystemClock.currentThreadTimeMillis()), new PushNotificationTrackingInfo(payload)));
        if (notificationManager != null) {
            try {
                notificationManager.notify(i, notification);
                return;
            } catch (SecurityException e) {
                DT.e("nf_push", "We are missing privilege?", e);
                return;
            } catch (Throwable th) {
                DT.e("nf_push", "Unexpected failure when trying to send notification", th);
                return;
            }
        }
        DT.b("nf_push", "Notification manager is not available!");
    }

    /* JADX INFO: Access modifiers changed from: package-private */
    @SuppressLint({"CheckResult"})
    public static void d(final Context context, final Payload payload, final InterfaceC6885pN interfaceC6885pN, final int i) {
        Payload.Action[] actions;
        Uri payload2;
        Objects.requireNonNull(context);
        Objects.requireNonNull(payload);
        Objects.requireNonNull(interfaceC6885pN);
        long when = payload.getWhen();
        String title = payload.getTitle(context.getString(com.netflix.mediaclient.ui.R.m.f773aa));
        String ticker = payload.getTicker(title);
        int i2 = com.netflix.mediaclient.ui.R.e.af;
        final Notification.Builder builder = new Notification.Builder(context);
        NotificationUtils.c(builder, context);
        builder.setContentIntent(d(context, payload, i));
        builder.setDeleteIntent(b(context, payload, i));
        builder.setTicker(ticker);
        builder.setAutoCancel(true);
        builder.setContentTitle(title);
        builder.setContentText(payload.text);
        builder.setSmallIcon(i2);
```

The information given by this section can be used to find what kind of operations are performed by an app, and to detect any unusual operation.

The browsable activities section reports, as the name suggests, a list of browsable activities exported by the app. These activities are activities that can be launched by links in the browser. For example, the first activity in the list can be launched by a webpage by using a link with a "nflx://" scheme and www.netflix.com as the host. This information can be used to find additional attack vectors: if one browsable activity contains a vulnerability, an attacker does not need to install a malicious app on the victim's device to launch it, but may just send a link via email.



o/C2269alO.java
o/C2935aut.java

Showing 1 to 10 of 29 entries

Previous 1 2 3 Next

BROWSABLE ACTIVITIES

Search:

ACTIVITY	INTENT
.acquisition.screens.signupContainer.SignupNativeDeepLinkActivity	**Schemes**: nflx://, **Hosts:** www.netflix.com, **Path Prefixes:** /confirmpageinfosignout, /loginfromregistration,
.ui.launch.NetflixComLaunchActivity	**Schemes**: http://, https://, nflx://, **Hosts:** www.netflix.com, app.netflix.com, msg.netflix.com, **Path Prefixes:** /profiles/icon, /Browse, /browse, /438c60eb, **Path Patterns:** /browse, /.*/browse, /browse/genre/.*, /.*/browse/genre/.*, /browse/coming-soon, /.*/browse/coming-soon, /browse/audio-description, /.*/browse/audio-description, /specials/.*, /Kids/title/.*, /Kids/character/.*, /game/.*, /.*/game/.*, /title/.*, /.*/title/.*, /watch/.*, /.*/watch/.*, /nmwatch/.*, /.*/nmwatch/.*, /clips/.*, /search, /search/.*, /.*/search, /.*/search/.*, /add/.*, /.*/add/.*, /download/.*, /.*/download/.*, /sync/.*, /.*/sync/.*, /profiles, /.*/profiles, /profiles/icon.*, /.*/profiles/icon.*, /mobilehelp, /like/.*, /.*/like/.*, /dislike/.*, /.*/dislike/.*, /notification/.*, /extras/.*, /extrasFeed, /extrasFeed/.*, /extrasfeed, /extrasfeed/.*, /remind-me/.*, /account/signup, /simpleSetup/upgradeWaiting, /hook/.*, /multimonth.*, /connection/.*, /bU4kXkYg/.*,

Showing 1 to 2 of 2 entries

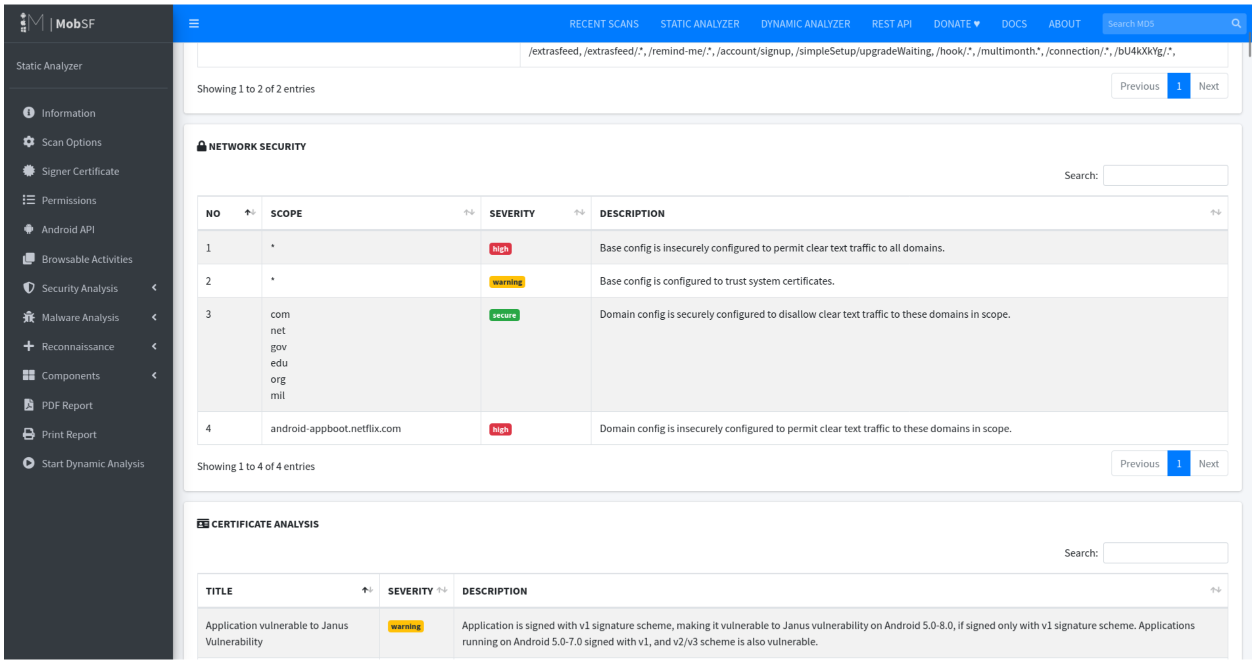
Previous 1 Next

NETWORK SECURITY

Search:

NO	SCOPE	SEVERITY	DESCRIPTION
1	*	high	Base config is insecurely configured to permit clear text traffic to all domains.

Next, we start with the Security Analysis sections. First of all, there is the Network Security, which contains information about the network configuration of the app. Here, we can see that the app is configured to allow non-encrypted traffic, which could expose it to man-in-the-middle attacks. We see that there are some exceptions, so domains ending with .com, .edu, … do not allow such traffic, but it is still possible that the app does perform unencrypted communication with other domains. In the case of app developers, this information can be used to ensure that no sensitive data is transmitted to such domains, and in case of attackers this information can be used to potentially intercept and edit the app's network traffic.



/extrasfeed, /extrasfeed/.*, /remind-me/.*, /account/signup, /simpleSetup/upgradeWaiting, /hook/.*, /multimonth.*, /connection/.*, /bU4kXkYg/.*,

Showing 1 to 2 of 2 entries

NETWORK SECURITY

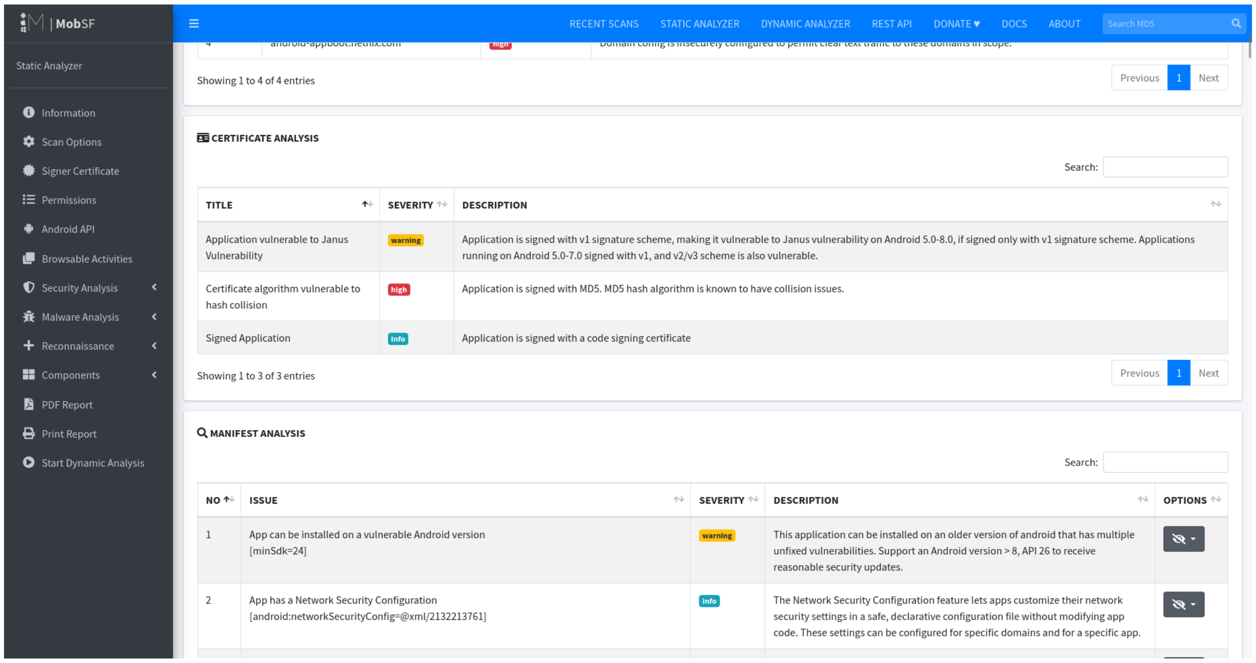
NO	SCOPE	SEVERITY	DESCRIPTION
1	*	high	Base config is insecurely configured to permit clear text traffic to all domains.
2	*	warning	Base config is configured to trust system certificates.
3	com net gov edu org mil	secure	Domain config is securely configured to disallow clear text traffic to these domains in scope.
4	android-appboot.netflix.com	high	Domain config is insecurely configured to permit clear text traffic to these domains in scope.

Showing 1 to 4 of 4 entries

CERTIFICATE ANALYSIS

TITLE	SEVERITY	DESCRIPTION
Application vulnerable to Janus Vulnerability	warning	Application is signed with v1 signature scheme, making it vulnerable to Janus vulnerability on Android 5.0-8.0, if signed only with v1 signature scheme. Applications running on Android 5.0-7.0 signed with v1, and v2/v3 scheme is also vulnerable.

While we previously saw all the information included in the app certificate, in the certificate analysis section MobSF performs some additional analysis regarding the security of the certificate. In this case, we can see that the app is signed, however it uses the md5 hash which is affected by collision issues. We can also see that the app could be vulnerable to the Janus vulnerability if installed on certain android versions.



4	android-appboot.netflix.com	high	Domain config is insecurely configured to permit clear text traffic to these domains in scope.

Showing 1 to 4 of 4 entries

Previous 1 Next

CERTIFICATE ANALYSIS

Search:

TITLE	SEVERITY	DESCRIPTION
Application vulnerable to Janus Vulnerability	warning	Application is signed with v1 signature scheme, making it vulnerable to Janus vulnerability on Android 5.0-8.0, if signed only with v1 signature scheme. Applications running on Android 5.0-7.0 signed with v1, and v2/v3 scheme is also vulnerable.
Certificate algorithm vulnerable to hash collision	high	Application is signed with MD5. MD5 hash algorithm is known to have collision issues.
Signed Application	info	Application is signed with a code signing certificate

Showing 1 to 3 of 3 entries

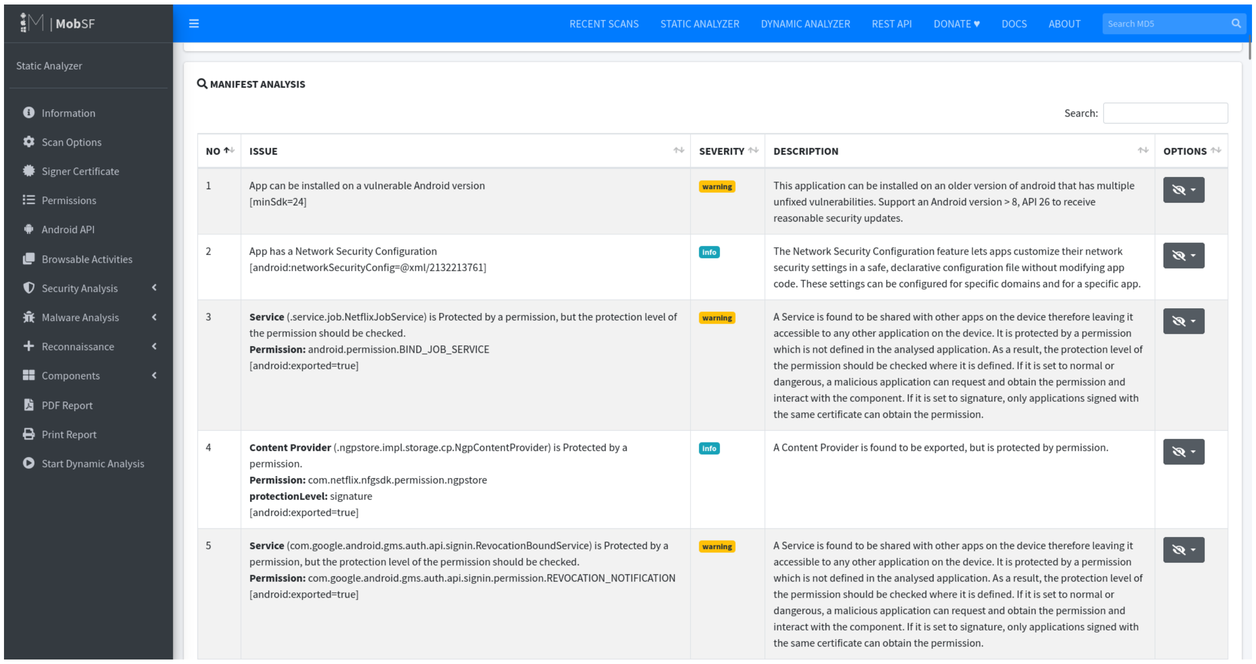
Previous 1 Next

MANIFEST ANALYSIS

Search:

NO	ISSUE	SEVERITY	DESCRIPTION	OPTIONS
1	App can be installed on a vulnerable Android version [minSdk=24]	warning	This application can be installed on an older version of android that has multiple unfixed vulnerabilities. Support an Android version > 8, API 26 to receive reasonable security updates.	
2	App has a Network Security Configuration [android:networkSecurityConfig=@xml/2132213761]	info	The Network Security Configuration feature lets apps customize their network security settings in a safe, declarative configuration file without modifying app code. These settings can be configured for specific domains and for a specific app.	

The next section, Manifest Analysis, includes all the information MobSF extracted from the manifest. For example, we can see that the app could be installed on old android devices, which are potentially vulnerable. We can also see information about the components exported by the app. For the NetflixJobService service, MobSF warns the user that while the service is protected, the permission it is protected with is not defined in the app itself, and so the user should check that the permission is defined at the appropriate protection level.

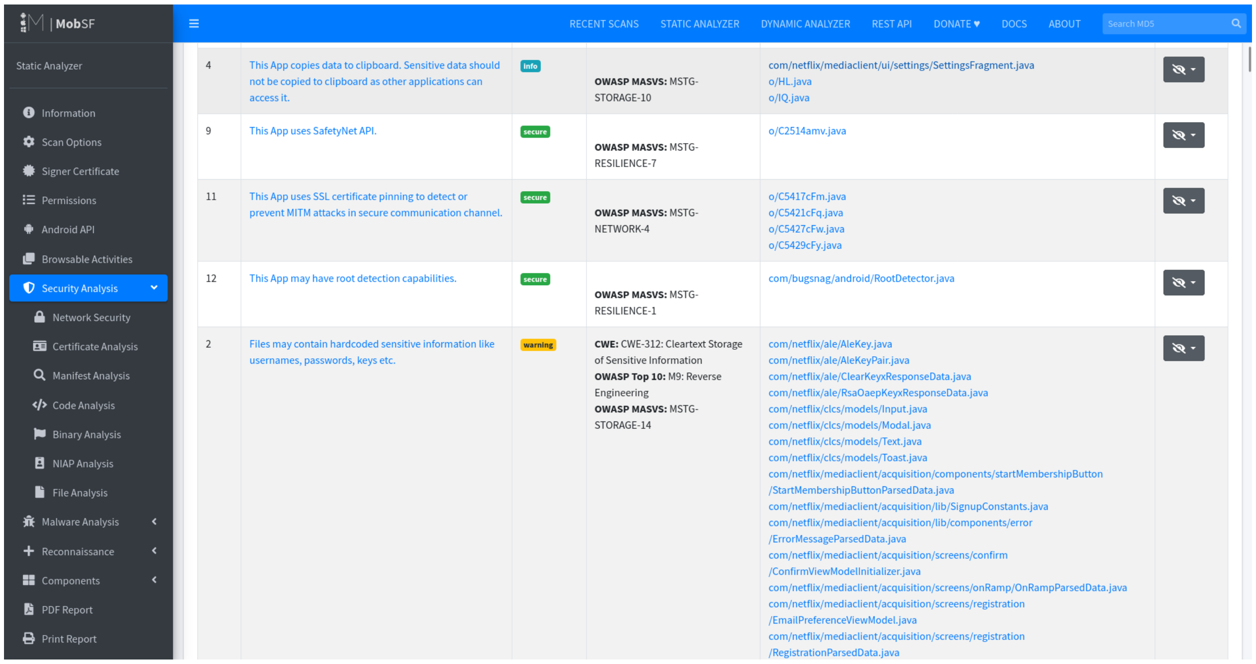


MANIFEST ANALYSIS

Search:

NO	ISSUE	SEVERITY	DESCRIPTION	OPTIONS
1	App can be installed on a vulnerable Android version [minSdk=24]	warning	This application can be installed on an older version of android that has multiple unfixed vulnerabilities. Support an Android version > 8, API 26 to receive reasonable security updates.	
2	App has a Network Security Configuration [android:networkSecurityConfig=@xml/2132213761]	info	The Network Security Configuration feature lets apps customize their network security settings in a safe, declarative configuration file without modifying app code. These settings can be configured for specific domains and for a specific app.	
3	**Service** (.service.job.NetflixJobService) is Protected by a permission, but the protection level of the permission should be checked. **Permission:** android.permission.BIND_JOB_SERVICE [android:exported=true]	warning	A Service is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device. It is protected by a permission which is not defined in the analysed application. As a result, the protection level of the permission should be checked where it is defined. If it is set to normal or dangerous, a malicious application can request and obtain the permission and interact with the component. If it is set to signature, only applications signed with the same certificate can obtain the permission.	
4	**Content Provider** (.ngpstore.impl.storage.cp.NgpContentProvider) is Protected by a permission. **Permission:** com.netflix.nfgsdk.permission.ngpstore **protectionLevel:** signature [android:exported=true]	info	A Content Provider is found to be exported, but is protected by permission.	
5	**Service** (com.google.android.gms.auth.api.signin.RevocationBoundService) is Protected by a permission, but the protection level of the permission should be checked. **Permission:** com.google.android.gms.auth.api.signin.permission.REVOCATION_NOTIFICATION [android:exported=true]	warning	A Service is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device. It is protected by a permission which is not defined in the analysed application. As a result, the protection level of the permission should be checked where it is defined. If it is set to normal or dangerous, a malicious application can request and obtain the permission and interact with the component. If it is set to signature, only applications signed with the same certificate can obtain the permission.	

Moving on, the code analysis section contains the information MobSF collected by analyzing the code of the app. MobSF reports all potential vulnerabilities it detected. In this case, if we sort by severity, we can see that the Netflix app has webview remote debugging enabled, which should be disabled for security reasons. We can also see that good security practices, such as certificate pinning, are highlighted as well.

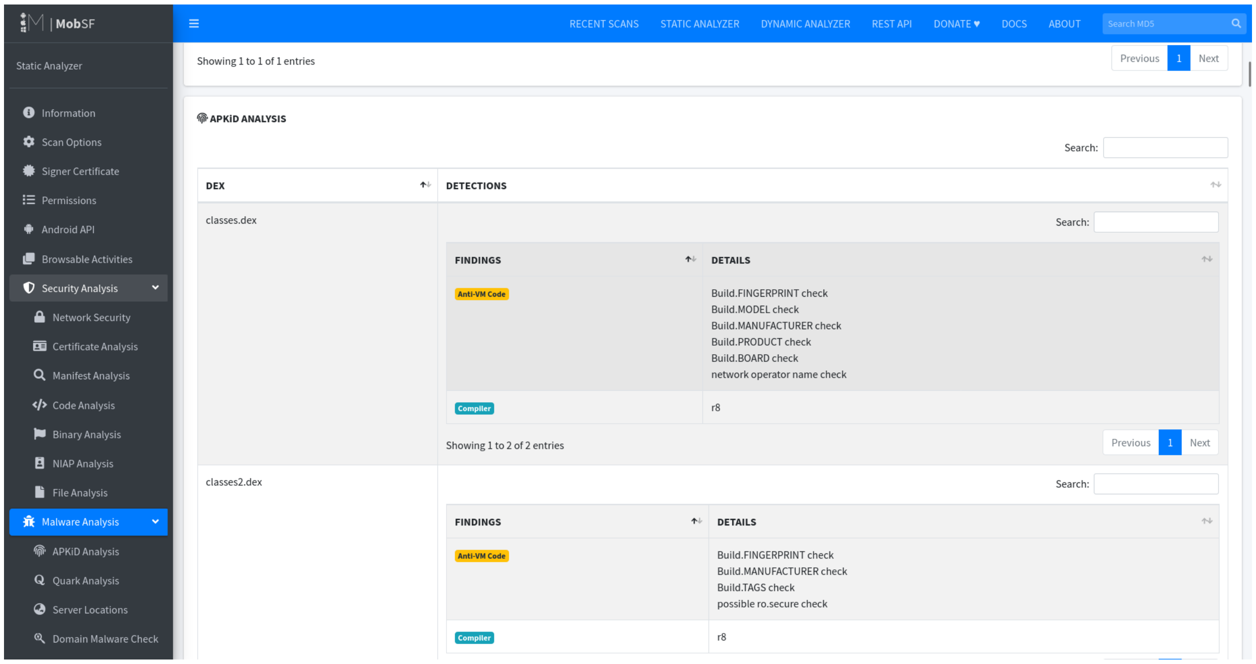


4	This App copies data to clipboard. Sensitive data should not be copied to clipboard as other applications can access it.	info	**OWASP MASVS:** MSTG-STORAGE-10	com/netflix/mediaclient/ui/settings/SettingsFragment.java o/HL.java o/IQ.java
9	This App uses SafetyNet API.	secure	**OWASP MASVS:** MSTG-RESILIENCE-7	o/C2514amv.java
11	This App uses SSL certificate pinning to detect or prevent MITM attacks in secure communication channel.	secure	**OWASP MASVS:** MSTG-NETWORK-4	o/C5417cFm.java o/C5421cFq.java o/C5427cFw.java o/C5429cFy.java
12	This App may have root detection capabilities.	secure	**OWASP MASVS:** MSTG-RESILIENCE-1	com/bugsnag/android/RootDetector.java
2	Files may contain hardcoded sensitive information like usernames, passwords, keys etc.	warning	**CWE:** CWE-312: Cleartext Storage of Sensitive Information **OWASP Top 10:** M9: Reverse Engineering **OWASP MASVS:** MSTG-STORAGE-14	com/netflix/ale/AleKey.java com/netflix/ale/AleKeyPair.java com/netflix/ale/ClearKeyxResponseData.java com/netflix/ale/RsaOaepKeyxResponseData.java com/netflix/clcs/models/Input.java com/netflix/clcs/models/Modal.java com/netflix/clcs/models/Text.java com/netflix/clcs/models/Toast.java com/netflix/mediaclient/acquisition/components/startMembershipButton/StartMembershipButtonParsedData.java com/netflix/mediaclient/acquisition/lib/SignupConstants.java com/netflix/mediaclient/acquisition/lib/components/error/ErrorMessageParsedData.java com/netflix/mediaclient/acquisition/screens/confirm/ConfirmViewModelInitializer.java com/netflix/mediaclient/acquisition/screens/onRamp/OnRampParsedData.java com/netflix/mediaclient/acquisition/screens/registration/EmailPreferenceViewModel.java com/netflix/mediaclient/acquisition/screens/registration/RegistrationParsedData.java

The binary analysis section contains information about the native libraries included in the apk file. In this case, there is no info here, probably because the netflix app does not use any native library. If it did, here we could find info about potential security risks related to the native libraries. For example, we could see if a library was compiled without the stack canary. This would be an indication that stack overflow exploits could be effective in attacking that library.

The NIAP analysis section just reports information regarding the NIAP certification, i.e. if the app respects what is prescribed by the NIAP guidelines or not.

Next, we can take a look at ApkID Analysis, the first section of the malware analysis. Here, we find some information about how the apk was made, as well as if there is some strange behavior, which we can use to detect potential malware. In this case, we can see that the app includes some anti-vm code, which means it may try to detect if it is being run inside an emulator. This information can help us during the dynamic analysis, since we can keep in mind that the app may act differently if run on an emulator or on a stock device.



Showing 1 to 1 of 1 entries

APKiD ANALYSIS

DEX	DETECTIONS
classes.dex	

FINDINGS	DETAILS
Anti-VM Code	Build.FINGERPRINT check Build.MODEL check Build.MANUFACTURER check Build.PRODUCT check Build.BOARD check network operator name check
Compiler	r8

Showing 1 to 2 of 2 entries

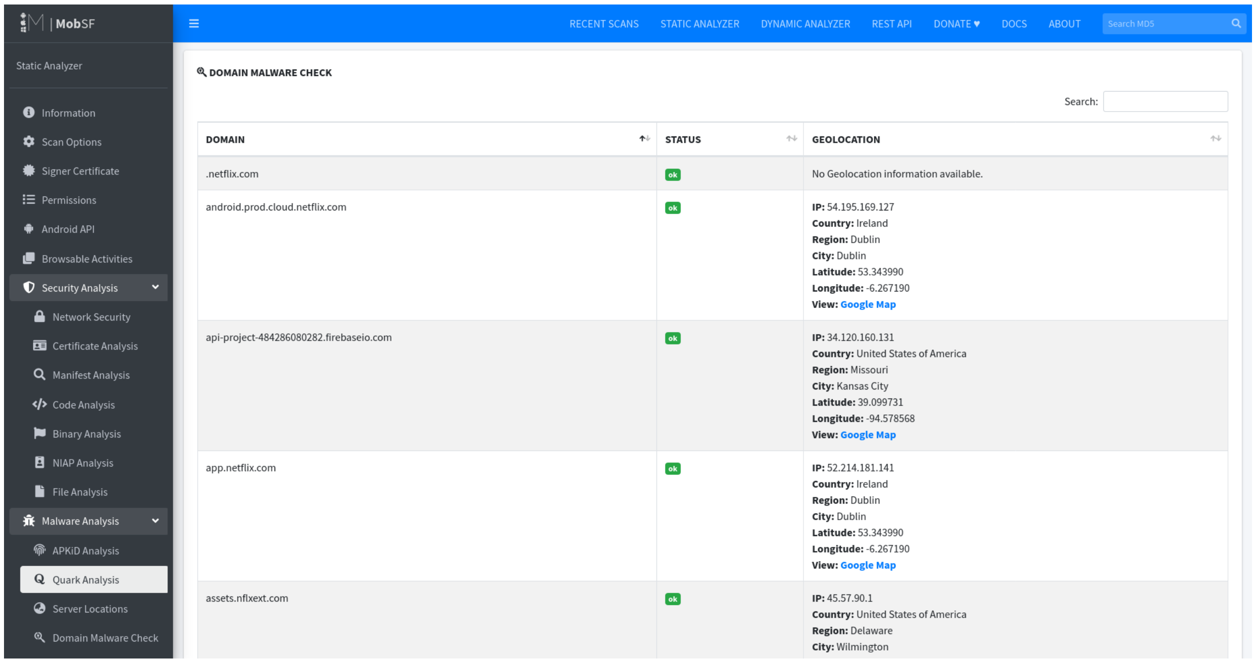
DEX	DETECTIONS
classes2.dex	

FINDINGS	DETAILS
Anti-VM Code	Build.FINGERPRINT check Build.MANUFACTURER check Build.TAGS check possible ro.secure check
Compiler	r8

The Quark Analysis section reports any potential malicious behavior detected by quark engine, another static analysis tool. In this case, no such behavior was found.

In the server location section, we see a map with the locations of the servers the app communicates with.

In the Domain Malware Check section, MobSF lists all the domains the app communicates with, checking if any of them is a known malware-related domain. As you can see, the netflix app does not have this kind of issue.



DOMAIN MALWARE CHECK

DOMAIN	STATUS	GEOLOCATION
.netflix.com	ok	No Geolocation information available.
android.prod.cloud.netflix.com	ok	**IP:** 54.195.169.127 **Country:** Ireland **Region:** Dublin **City:** Dublin **Latitude:** 53.343990 **Longitude:** -6.267190 **View:** Google Map
api-project-484286080282.firebaseio.com	ok	**IP:** 34.120.160.131 **Country:** United States of America **Region:** Missouri **City:** Kansas City **Latitude:** 39.099731 **Longitude:** -94.578568 **View:** Google Map
app.netflix.com	ok	**IP:** 52.214.181.141 **Country:** Ireland **Region:** Dublin **City:** Dublin **Latitude:** 53.343990 **Longitude:** -6.267190 **View:** Google Map
assets.nflxext.com	ok	**IP:** 45.57.90.1 **Country:** United States of America **Region:** Delaware **City:** Wilmington

Finally, there are the reconnaissance and the components sections. In reconnaissance, MobSF lists some strings that could be of interest during the analysis: urls, firebase databases, trackers, email addresses, and potentially hardcoded secrets. This section also reports all the strings included in the apk file.

In components, MobSF simply lists all the components defined by the application, divided by type, as well as the third party libraries used by the app.