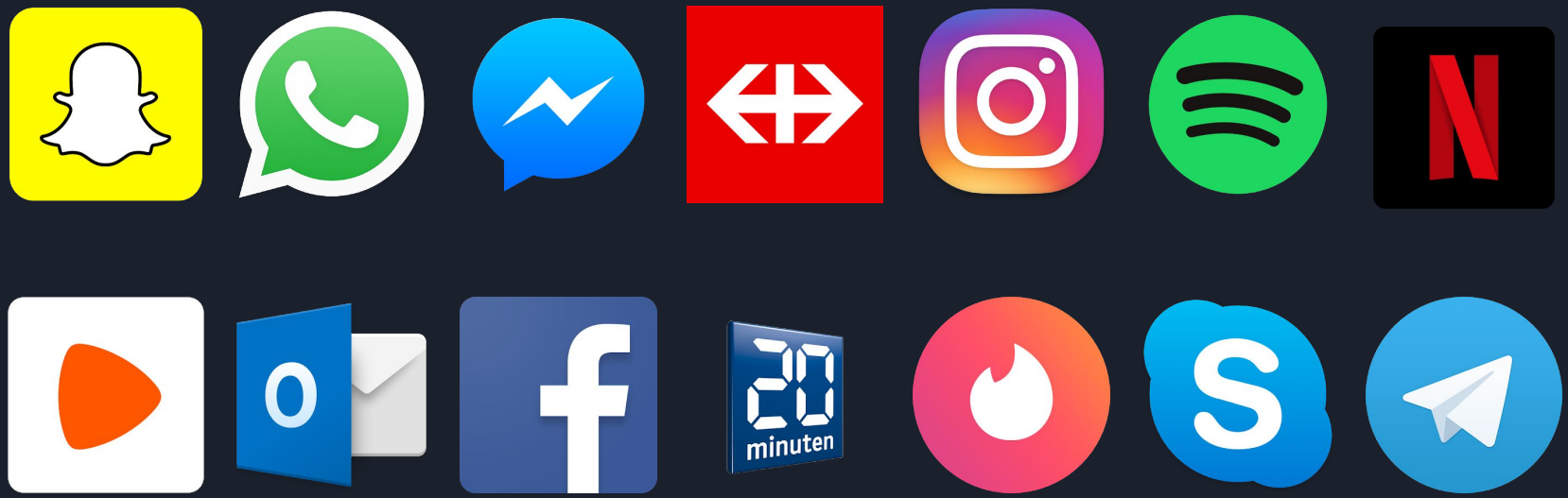
Targeted Attacks Supported by Recovered Data Structures of Web API's

Master Thesis - Introductory Presentation
Marc-Andrea Tarnutzer
23.10.2018

UNIVERSITÄT BERN

Networking Appified World



Problems

1. Software quality
   Sloppy developers
   Weak quality control

2. Software protection
   Code availability
   Endpoints / credentials exposure

3. App communications
   Sensitive data
   Flawed inputs

These Problems can lead to...

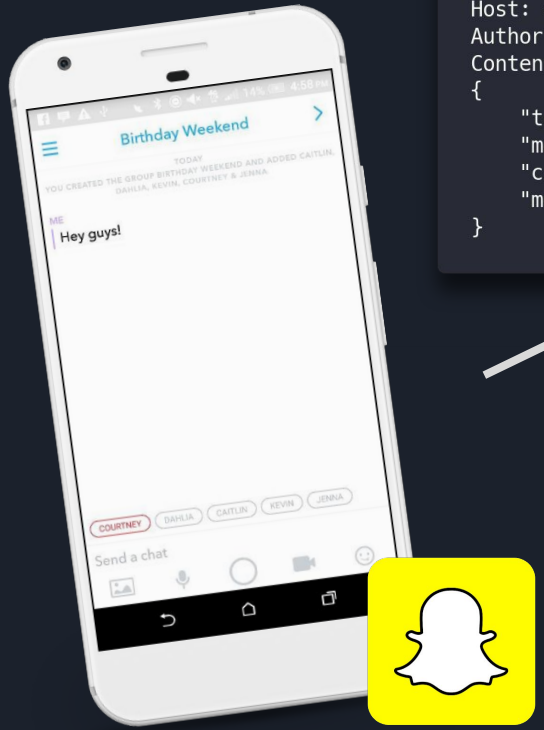
Data leaks / theft

Denial of service

Impersonation / Unauthorized billing

Example: Snapchat

```
POST / HTTP/1.1
Host: api.snapchat.com
Authorization: Bearer eyJhbGciUxMiJ9.eyJzdWIiNTI1MTIyMj
Content-Type: application/json
{
    "to-id":"94857892",
    "message-type":"text",
    "created_at":"2018-09-18T13:32:39.708Z",
    "message": "Hey guys!"
}
```



snapchatAPI
Full Snapchat API
Python ★ 27 13

Snapchat-SDK-PHP
PHP SDK for Snapchat
PHP

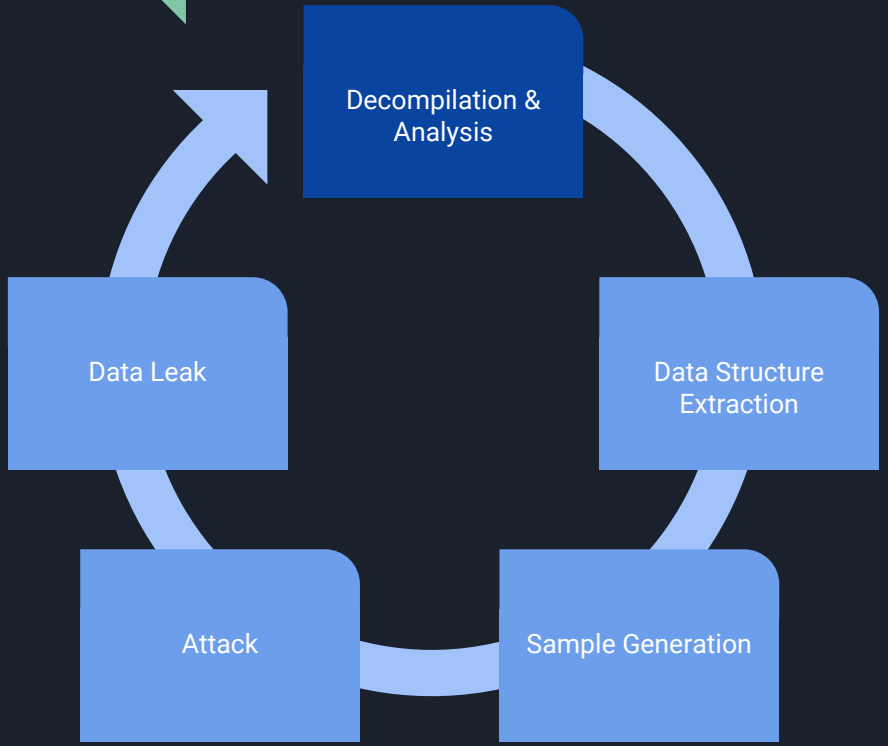
SC-API
PHP library of Snapchat's private API
PHP ★ 292 88

SnapchatKit
An Objective-C implementation of the unofficial Snapchat API.
Objective-C ★ 117 20

Goals

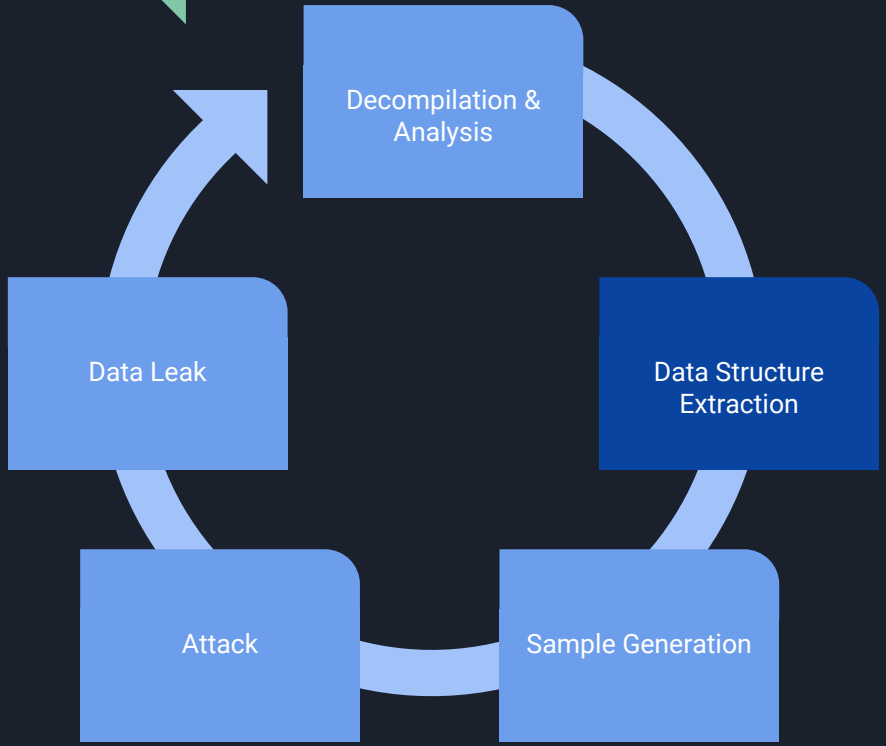
Risk Assessment	Automated App Analysis	Raise Awareness
Assess potential risks caused by client-server implementations	Automated analysis of Android applications	Raise awareness of encountered threats

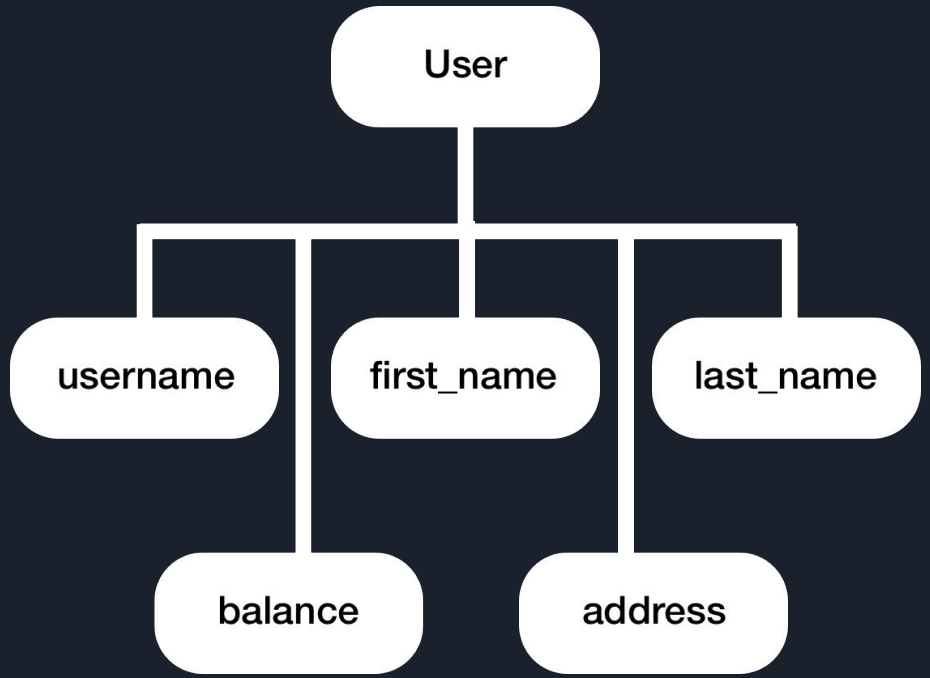
Automated App Analysis



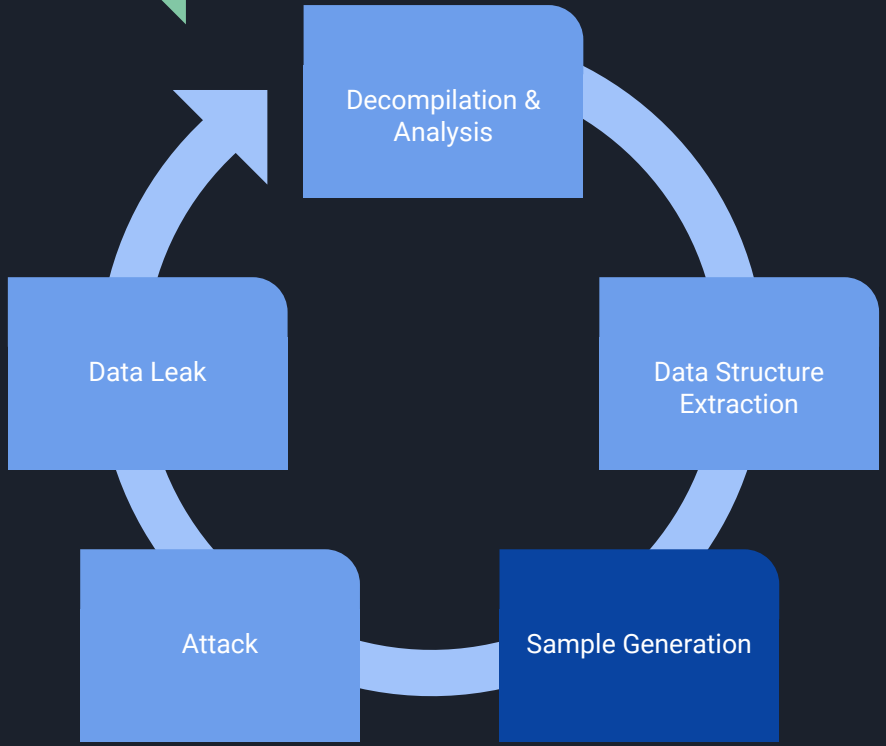

```
OkHttpClient client = new OkHttpClient();
Request request = new Request.Builder()
        .url(reqUrlString)
        .get().build();

client.newCall(request).enqueue(new Callback() {
…
@Override
public void onResponse(Call call,
        Response response) throws IOException {
        String response = response.body().string();
        JSONObject jObj = new JSONObject(response);
        User user = new User(
                jObj.getString(“username”),
                jObj.getString(“first_name”),
                jObj.getString(“last_name”),
                jObj.getDouble(“balance”),
                jObj.getString(“address”));
        }
});
```

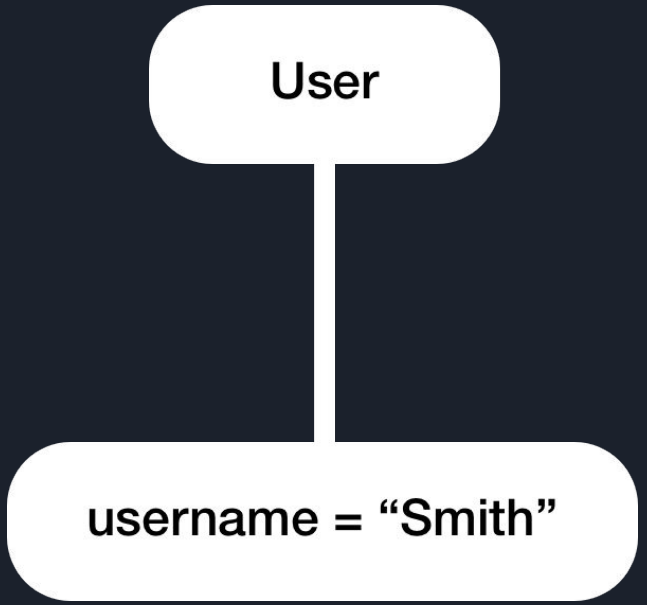
Automated App Analysis



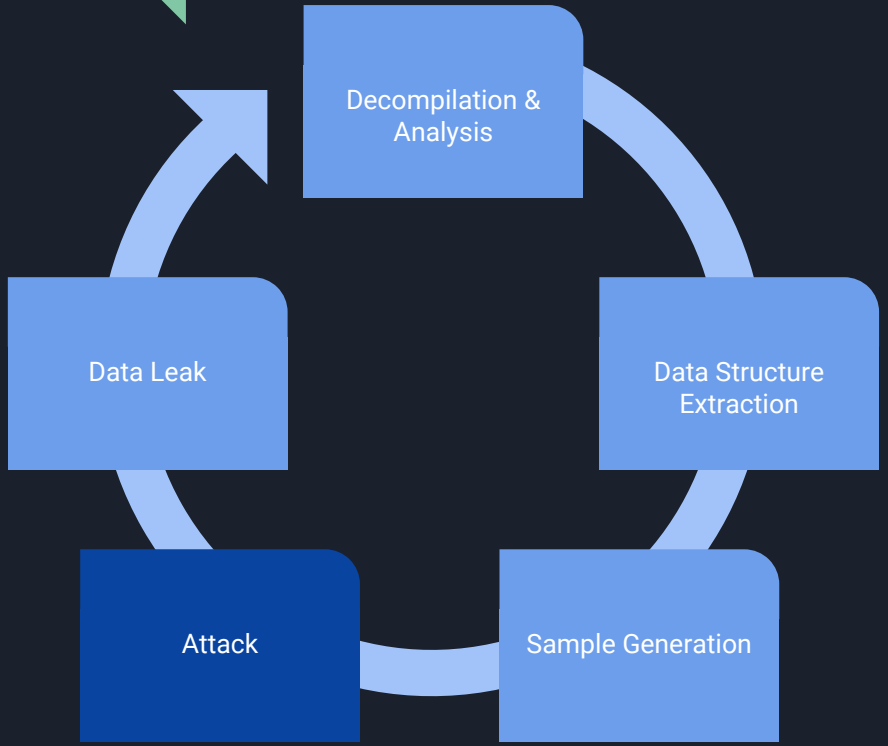


Automated App Analysis





Automated App Analysis

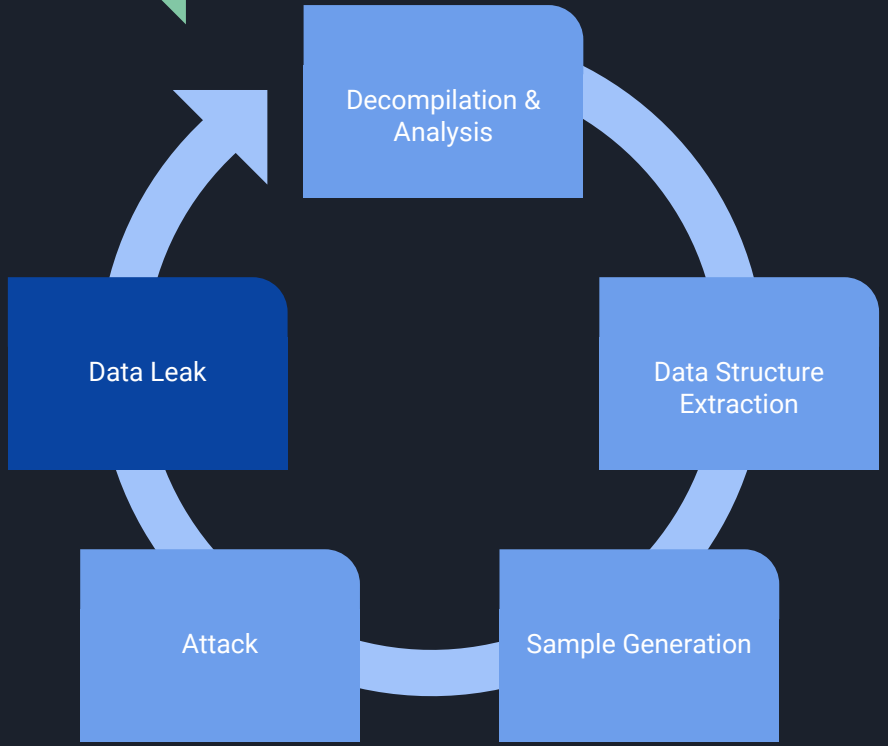


https://bankname.com/
getDetails?username=Smith

- or -

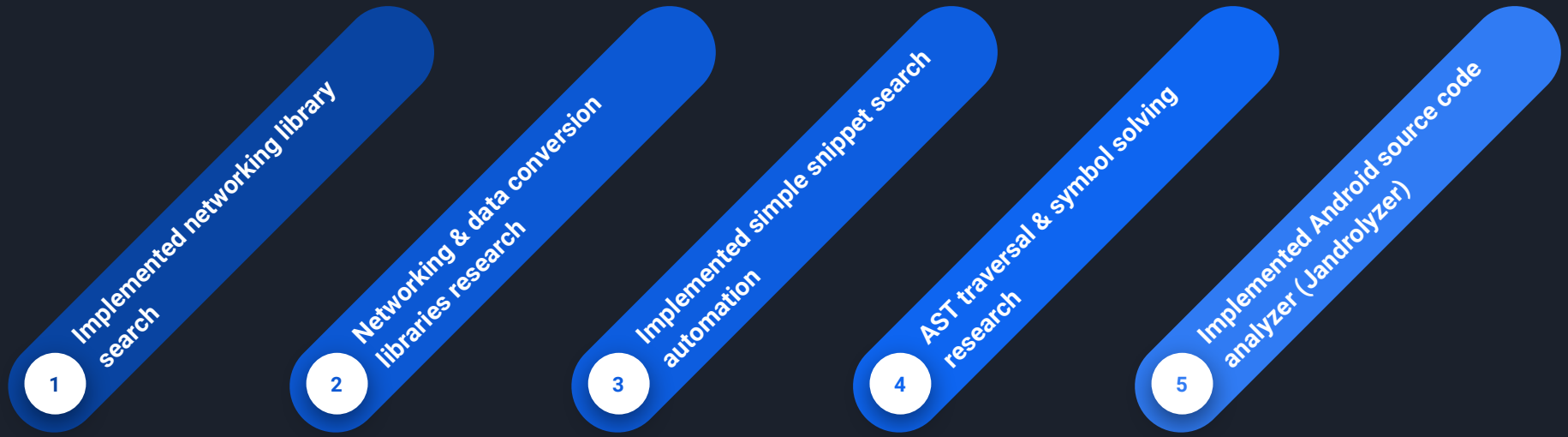
predicted JSON object in header

Automated App Analysis

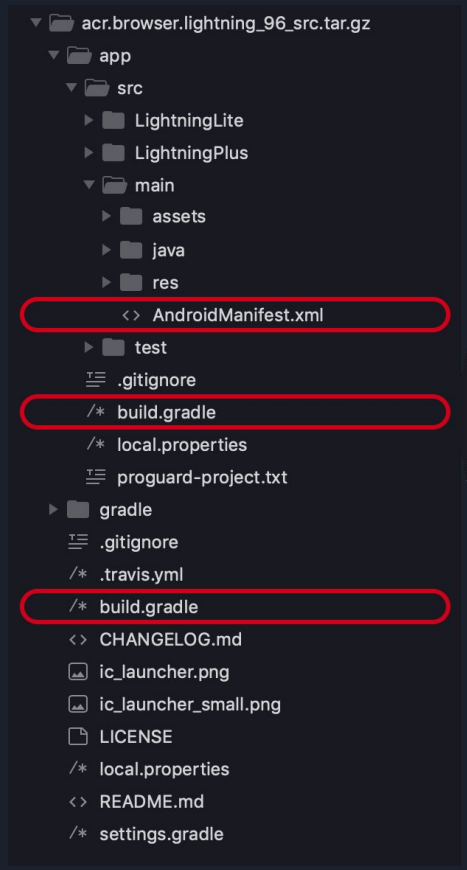


{
“username”: “Smith”,
“first_name”: “Mike”,
“last_name”: “Smith”,
“balance”: 9999999,
“address”: “767 5th Ave
New York, NY 10153”
}

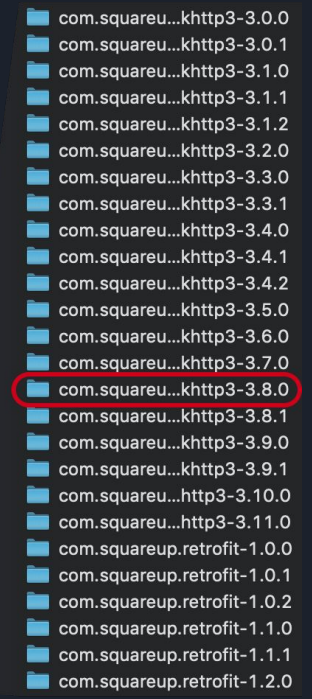
Progress


1
Implemented networking library search
2
Networking & data conversion libraries research
3
Implemented simple snippet search automation
4
AST traversal & symbol solving research
5
Implemented Android source code analyzer (Jandrolyzer)

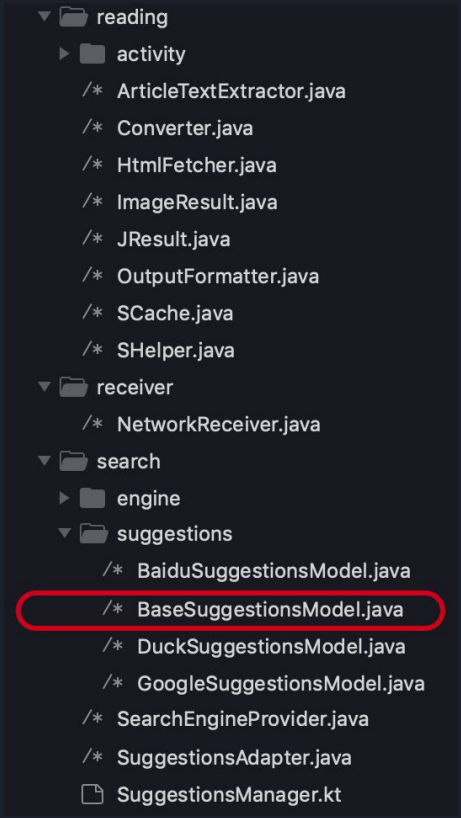
Jandrolyzer: Library Support

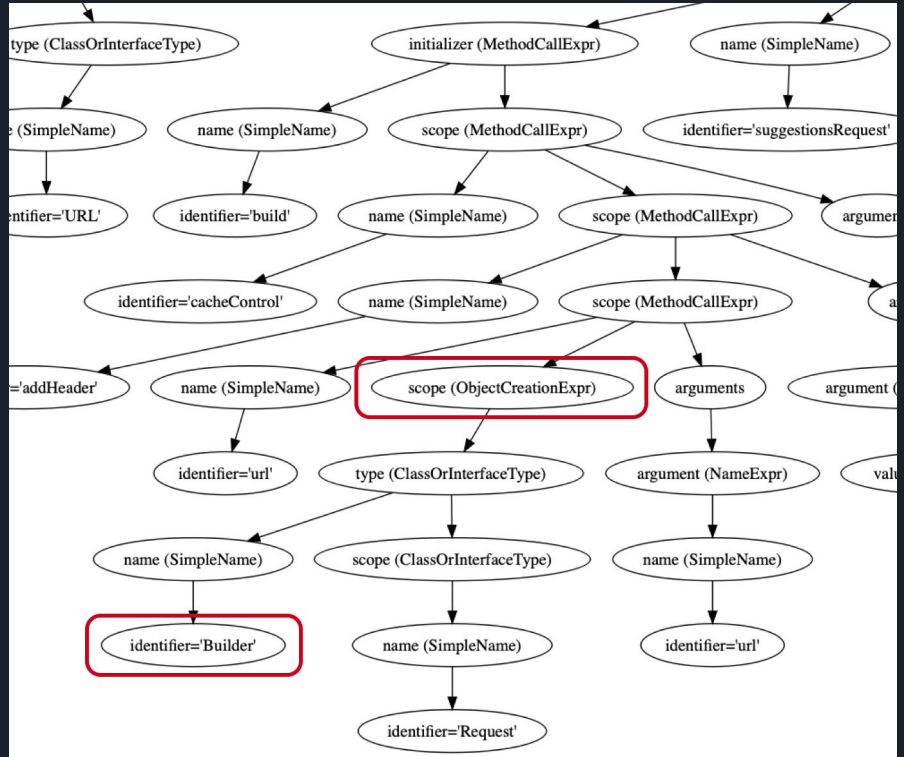



```
dependencies {
    testCompile 'junit:junit:4.12'

    // support libraries
    def supportLibVersion = '25.4.0'
    compile "com.android.support:palette-v7:$supportLibVersion"
    compile "com.android.support:appcompat-v7:$supportLibVersion"
    compile "com.android.support:design:$supportLibVersion"
    compile "com.android.support:recyclerview-v7:$supportLibVersion"
    compile "com.android.support:support-v4:$supportLibVersion"

    // html parsing for reading mode
    compile 'org.jsoup:jsoup:1.10.2'

    // dependency injection
    def daggerVersion = '2.11'
    compile "com.google.dagger:dagger:$daggerVersion"
    annotationProcessor "com.google.dagger:dagger-compiler:$daggerVersion"
    provided 'javax.annotation:jsr250-api:1.0'

    // view binding
    def butterknifeVersion = '8.6.0'
    compile "com.jakewharton:butterknife:$butterknifeVersion"
    annotationProcessor "com.jakewharton:butterknife-compiler:$butterknifeVersion"

    // permissions
    compile 'com.anthonycr.grant:permissions:1.1.2'

    // proxy support
    compile 'net.i2p.android:client:0.8'

    compile 'com.squareup.okhttp3:okhttp:3.8.0'
```



Jandrolyzer: AST Traversal





ObjectCreationExpr of type: okhttp3.Request.Builder

Jandrolyzer: Networking Snippet Example

```
@Nullable
private InputStream downloadSuggestionsForQuery(@NonNull String query, @NonNull String language) {
    String queryUrl = createQueryUrl(query, language);

    try {
        URL url = new URL(queryUrl);

        // OkHttp automatically gzips requests
        Request suggestionsRequest = new Request.Builder().url(url)
            .addHeader("Accept-Charset", mEncoding)
            .cacheControl(mCacheControl)
            .build();

        Response suggestionsResponse = mHttpClient.newCall(suggestionsRequest).execute();

        ResponseBody responseBody = suggestionsResponse.body();
        return responseBody != null ? responseBody.byteStream() : null;
    } catch (IOException exception) {
        Log.e(TAG, "Problem getting search suggestions", exception);
    }

    return null;
}
```

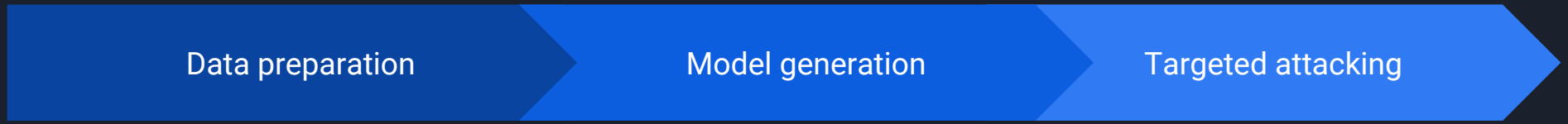
Challenges

Memory

JavaParser expression resolving

Implementation variations

Next Steps



Decompiled APK analysis

Data structure extraction

Endpoint extraction

Property evaluation

Sample generation

Automated endpoint testing

Summary

Problems

Goals

Automated App Analysis

Progress

Jandrolyzer

Challenges

Next Steps