

# VBA Native Android Onboarding

Security Architecture Engineering

Exported on 04/13/2022

## Table of Contents

1	Visa Confidential .....	3
2	Release notes .....	4
2.1	1.0.2.....	4
2.2	1.0.1.....	4
2.3	1.0.0.....	4
3	Loading the Agent.....	5
4	Initializing the Agent and Begin Profiling .....	6
4.1	Import VBA.....	6
5	Response Format .....	8
6	Environments .....	9
6.1	Visa Confidential .....	9

# 1 Visa Confidential

Current VBA Android SDK version: **1.0.2**

Minimum supported Android API: **26 (Android 8)**

## 2 Release notes

### 2.1 1.0.2

Added India localization support. See notes on environments below.

### 2.2 1.0.1

Improved runtime hook detection and root detection

### 2.3 1.0.0

Initial release

### 3 Loading the Agent

The VBA Android SDK is provided as a local Maven repository which contains an AAR. To add this dependency to your project:

1. Download the latest SDK (contact [GDLSecurityEngineeringVBA@visa.com](mailto:GDLSecurityEngineeringVBA@visa.com)<sup>1</sup> for download link)
2. Extract the tarball which contains a local Maven repository:

```
$ tar -xvzf VBA-Android-<version>.tar.gz
```

3. Add the local Maven repository to your **project** build.gradle. This will look like:

```
allprojects {
    repositories {
        ...
        maven {
            url "/path/to/extracted/vba-maven-repository"
        }
    }
}
```

4. Add VBAAgent as an **implementation** dependency in your **application** build.gradle. This will look like:

```
dependencies {
    ....
    implementation ('com.visa:VBAAgent:<version>@aar'){transitive=true}
}
```

Note that `transitive=true` ensures that transitive VBA dependencies are included. In case there are runtime conflicts such as duplicate classes detected, you may have to exclude one.

---

<sup>1</sup> <mailto:GDLSecurityEngineeringVBA@visa.com>

## 4 Initializing the Agent and Begin Profiling

Prior to initiating critical, protected flow please call VBA agent's profiling function to trigger offline token delivery and attestation calculation. The steps to do so are as follows:

1. Import VBA
2. Initialize a VBAAgentConfig object with:
  - a. **Session ID:** An ID (e.g. UUID) to identify the session, track this as it will be used from VBA to bind attestation results
  - b. **Service Name:** Application/Service Name
  - c. **Merchant ID:** Merchant Name
  - d. **Environment:** VBA currently supports two environments, Dev and Prod. The SDK exposes both as a public enum. Either use VBAEnvironment.DEVELOPMENT or VBAEnvironment.PRODUCTION
3. Initialize the VBA Agent with the config
4. Call VBA Agent profile function with a completion handler. See code below:

### 4.1 Import VBA

```
import com.visa.app.vbaagent.core.OnVBAAgentEventListener;
import com.visa.app.vbaagent.core.VBAAgent;
import com.visa.app.vbaagent.core.VBAAgentConfig;
import com.visa.app.vbaagent.core.VBAEnvironment;

// Sample implementation in a Fragment, but VBA Agent can also be initialized anywhere
public class LoginFragment extends Fragment {

    @Override
    public void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        ...
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        ...

        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String sessionId = UUID.randomUUID().toString();
                String service = "DemoService";
                String merchantId = "DemoMerchant";

                // Initialize agent
                VBAAgentConfig config = new VBAAgentConfig(sessionId, service, merchantId,
                    VBAEnvironment.DEVELOPMENT);
                VBAAgent agent = new VBAAgent(context, config);

                // Begin profiling
                agent.profile(new OnVBAAgentEventListener() {
                    @Override
```

```
        public void onSuccess() {  
            // Continue with API flow  
            ...  
        }  
  
        @Override  
        public void onError(Exception e, int errorCode) {  
            // Continue with API flow, and log VBA error with session ID  
            ...  
        }  
    }  
};  
}  
};  
...  
}  
}
```

## 5 Response Format

### Completion Handler Listeners

- onSuccess() - indicates successful token delivery
- onError(Exception e, int errorCode) - indicates an internal VBA failure, please log this response with session ID
  - e - the exception
  - errorCode - code of the exception (see below)

### Error Response Codes:

- 400: Request failed, typically indicates network outage or unstable internet connection
- 500: VBA Server error
- 0: VBA SDK Error



## 6 Environments

There are three environments available under the VBAEnvironment enum:

VBAEnvironment.DEVELOPMENT: use this environment for testing and other non-production traffic.

VBAEnvironment.PRODUCTION\_US: use this environment if your customer base is located **outside India**

VBAEnvironment.PRODUCTION\_INDIA: use this environment if your customer base is located **within India**

Each of these environments are mutually exclusive and do not share any data. If you are not sure which environment to use, please contact [GDLSecurityEngineeringVBA@visa.com](mailto:GDLSecurityEngineeringVBA@visa.com)<sup>2</sup>.

### 6.1 Visa Confidential

---

<sup>2</sup> <mailto:GDLSecurityEngineeringVBA@visa.com>