What is Android Root Checker?

This is easy-to-use asset, which allows you to check if Android device is rooted.

Root can be used to emulate in-app purchases and/or use cheat engines by installing software to affect system logic and security mechanisms. All you need is to invoke one method, which returns true or false - is device rooted or not:

AndroidRootChecked.IsRooted()

This works well on Personal/Plus/Pro/Enterprise Unity, suitable for Android with any firmwares, that Unity supports.

How does it works?

Asset has seven checks - it checks installed root packages, files, binaries, dangerous props, read/write permissions, test keys and root commands which are present on rooted device. If Android device is rooted, **AndroidRootChecker.IsRooted()** returns true, otherwise method returns false.

AndroidRootChecker class

AndroidRootChecker is static class that contains methods for checking is device rooted.

AndroidRootChecker class methods:

`public static bool CheckPackages()` - returns if device has installed root packages;

`public static bool CheckFiles()` - returns if device has root files;

`public static bool CheckBinaries()` - returns if device has root binaries;

`public static bool CheckDangerousProps()` - returns if device has root dangerous props;

`public static bool CheckReadWritePaths()` - returns if device has root read/write permissions on specific folders;

`public static bool CheckTestKeys()` - returns if device has root test keys;

`public static bool CheckCommandsExists()` - returns if device has root commands;

`public static bool IsRooted()` - return if device is rooted using all checks.

Please let me know if you have any questions.
E-mail: `unitymedved@gmail.com`