Severity
High

VULNERABILITY

Unsafe TrustManager implementation

OWASP MASVS

5.3 L2

Common Weakness Enumeration

CWE-295

Known Exploits

CVE-2020-5523

Common Vulnerability Scoring System

CVE-2020-5523-CVSS 3.0 Score 7.4

Best Practices:

https://github.com/OWASP/owasp-mstg/blob/1.1.3/Document/0x05g-Testing-Network-Communication.md#testing-endpoint-identify-verification-mstg-network-3

Reference URL[s]:

https://developer.android.com/reference/javax/net/ssl/TrustManager
https://find-sec-bugs.github.io/bugs.htm#WEAK_TRUST_MANAGER
https://support.google.com/faqs/answer/6346016?hl=en
https://stackoverflow.com/questions/35545126/an-unsafe-implementation-of-the-interface-x509trustmanager-from-google
https://www.securecoding.cert.org/confluence/pages/viewpage.action?pageId=134807561
https://www.jssec.org/dl/android_securecoding_en.pdf

CVSS BaseScore and Vector

CVE-2020-5523
Version & Base Score : CVSS 3.0 Score 7.4
CVSS Scoring Vector :
CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N

THREAT

TrustManagers are responsible for managing the trust material that is used when making trust decisions and for deciding whether credentials presented by a peer should be accepted

RISK

The TrustManager interface might have been configured to trust all the server certificates, regardless of who signed it

An implementation ignoring all the SSL certificate validation errors when establishing an HTTPS connection to a remote host makes your app vulnerable to MitM attacks

Beginning 17 May 2016, Google Play started to block publishing any new apps or updates containing an unsafe implementation of the interface X509TrustManager

FIX

To properly handle SSL certificate validation, change your code in the checkServerTrusted method of your custom X509TrustManager interface to raise either CertificateException or IllegalArgumentException whenever the certificate presented by the server does not meet your expectations

We have detected 'X509Certificate[] getAcceptedIssuers()' in the file org/jivesoftware/smack/util/TLSUtils.java

line : 20 import javax.net.ssl.SSLSocket;

line : 21 import javax.net.ssl.X509TrustManager;

line : 22 import org.jivesoftware.smack.ConnectionConfiguration;

line : 49 @Override

line : 50 public X509Certificate[] getAcceptedIssuers() {

line : 51 return new X509Certificate[0];

We have detected 'X509Certificate[] getAcceptedIssuers()' in the file org/minidns/dane/ExpectingTrustManager.java

line : 4 import java.security.cert.X509Certificate;

line : 5 import javax.net.ssl.X509TrustManager;

line : 6

line : 43 @Override

line : 44 public X509Certificate[] getAcceptedIssuers() {

line : 45 return this.trustManager.getAcceptedIssuers();

Severity
Medium

VULNERABILITY

Debugging Information Provision

OWASP MASVS

7.4 L2

Common Weakness Enumeration

CWE-215

Known Exploits

CAPEC-133

CVE-2018-6599

Common Vulnerability Scoring System

CVE-2018-6599-CVSS 3.0 Score 5.5

Reference URL[s]:

https://github.com/b66l/OASAM/blob/master/oasam-leak-information-leak/oasam-leak-002-information-leak-to-log-files.md

CVSS BaseScore and Vector

CAPEC-133

CVE-2018-6599
Version & Base Score : CVSS 3.0 Score 5.5
CVSS Scoring Vector : CVSS:3.0/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N

THREAT

Applications can output runtime information using the android.util.Log class. Logcats are usually collected [e.g. via adb logcat -v long > logcat.txt] in order to bugfix the app but they should be removed once done