Android OSは安全なのか？

GDG DevFest Tokyo 2018
2018/9/1(土) 10:00〜10:30
RoomI セッション1

タオソフトウェア株式会社
代表取締役
谷口岳
@tao_gaku

RiskFinder

自己紹介

タオソフトウェア株式会社

- 代表取締役　谷口岳
- ２００５年創業、Android関連システム受託開発
- Google Playにアプリを多数公開
- ブログにて開発者向け情報を発信（昔）
- 雑誌執筆、講演
- セキュリティ色々

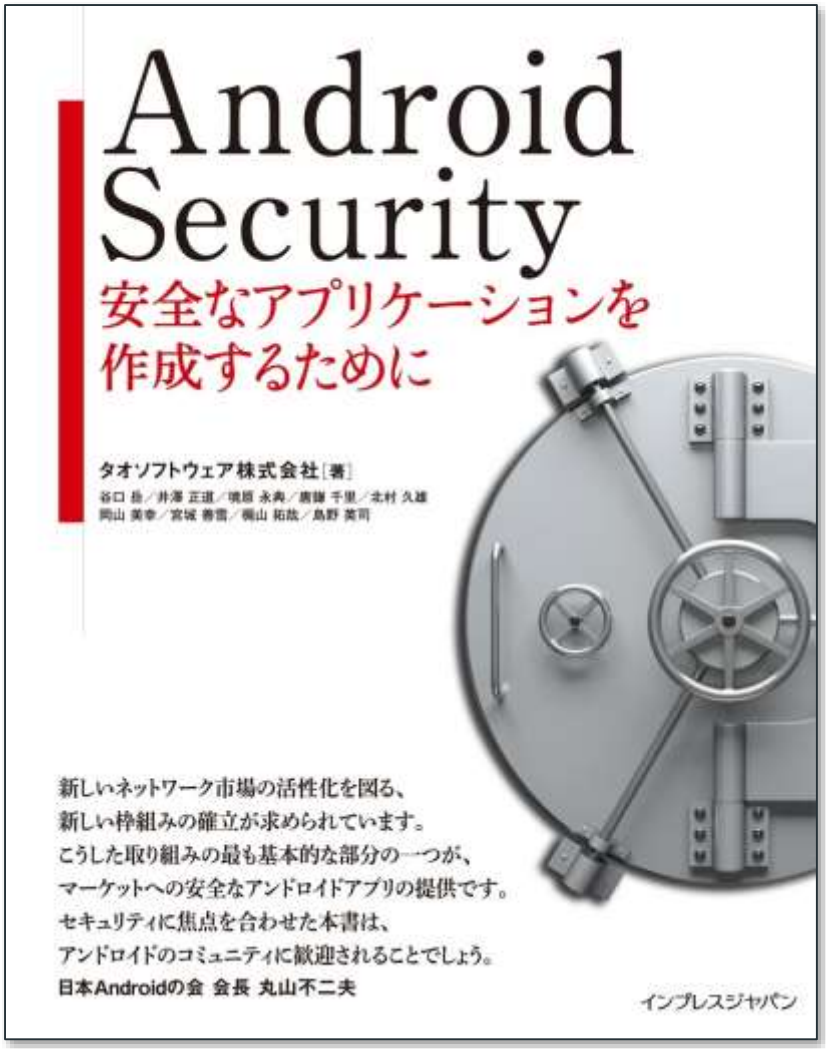
自己紹介

タオバイザー

3DVRゴーグル

- クラウドファンディングで資金調達
- http://taovisor.com

Android Security 安全なアプリケーションを作成するために



２０１２年１月１日発刊

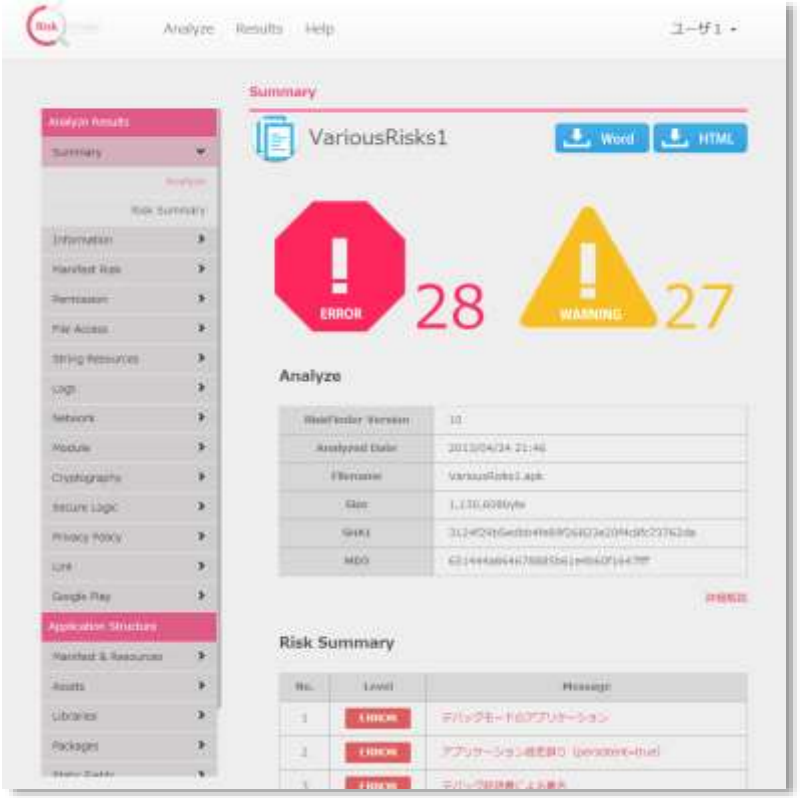
開発者向け

出版社：インプレスジャパン

RiskFinder　アンドロイドアプリの脆弱性発見ツール

ＡＰＫファイルをアップロードするだけで脆弱性レポートが作成されます。

講演をする中で、
「気を付ける事が沢山あるのは分かった。
でも全てのプログラマが理解するのは
難しい何かいい方法はないか？」
という声があったので作ってみました。



1. プログラマでなくても使える
2. ソースコード不要
3. ウエブサービス型
4. 脆弱性以外も検出

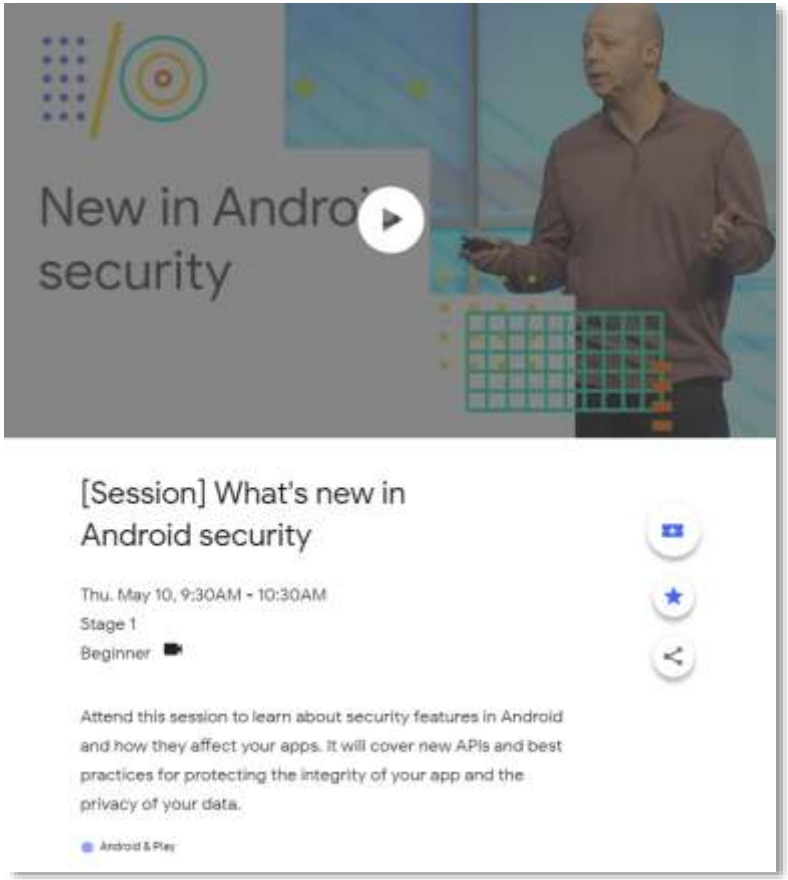
本セッションについて

Android OSは、昔は非常に危険なOSでしたが、年々バージョンアップを繰り返し、多くのセキュリティ課題をクリアしてきました。

Google I/O 2018の「What is new Android P Security」の講演内容をベースに現在のアンドロイドの安全性について解説をします。

またAndroid Pでのセキュリティに関する事項を解説致します。

What’s new in Android Security Session



Google I/O 2018

5月10日(木) 9:30～

Youtube:
https://www.youtube.com/watch?time_continue=12&v=r54roADX2MI

Androidは安全である

Androidのセキュリティは他のモバイルプラットフォームと同じレベルである。

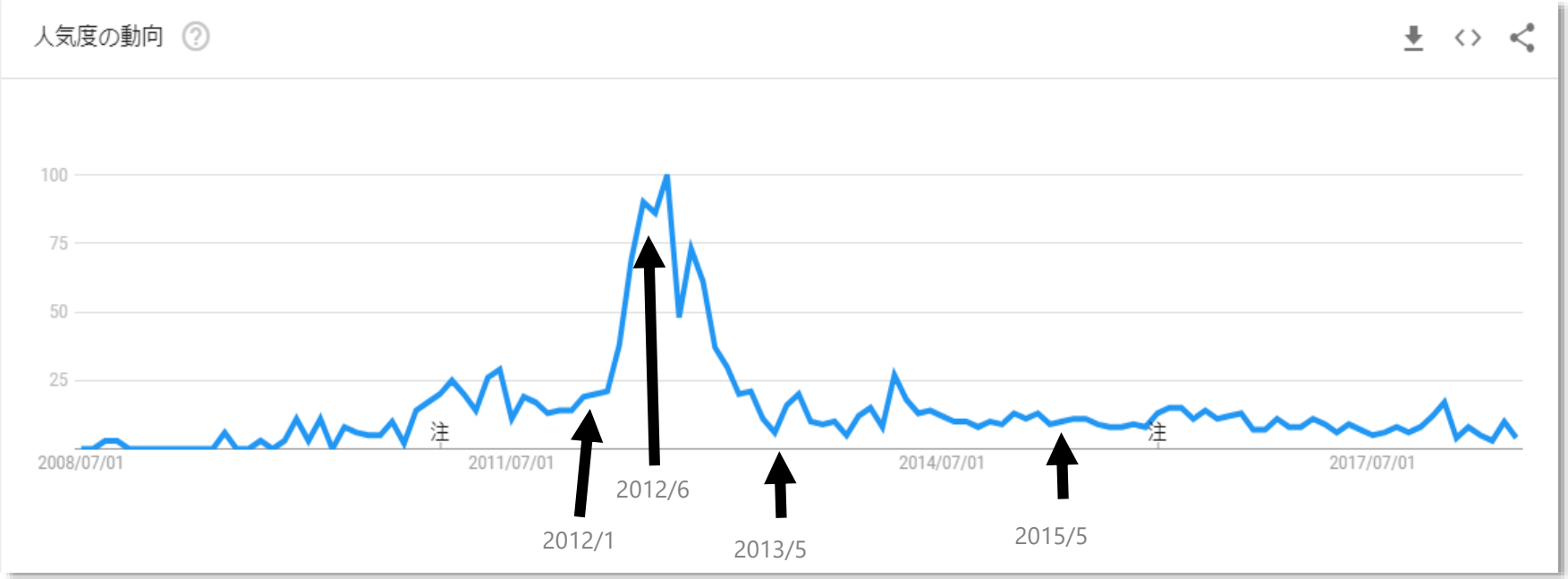
by Google

Androidセキュリティ情報

- 2012/1 Android Security(安全なアプリケーションを作成するために）(タオソフトウエア)
- 2012/6 Androidアプリのセキュア設計・セキュアコーディングガイド(JSSEC)
- 2012/6 IPAテクニカルウオッチ(IPA)
- 2012/7 スマートフォンプライバシーイニシアチブ(総務省)
- 2015/5 Androidアプリの脆弱性の学習・点検ツール AnCole（IPA）

Android セキュリティの注目度

セキュリティ情報が提供されず最新のセキュリティ情報が世の中にアップデートされていない



- 「Android セキュリティ」のGoogle Trands結果
- 期間:2008/8～2017/7
- ピーク2012/8
- Google Trends: https://trends.google.co.jp/

3つの証明？

1 Transparency（透明性）

Measurability（測定可能性）

- 2．PHAをインストールした発生率を見る
- 3．エクスプロイトコード価格を見る

1. Transparency

AndroidOSが安全である理由1

Transparency（透明性）＝オープン

Androidプラットフォーム

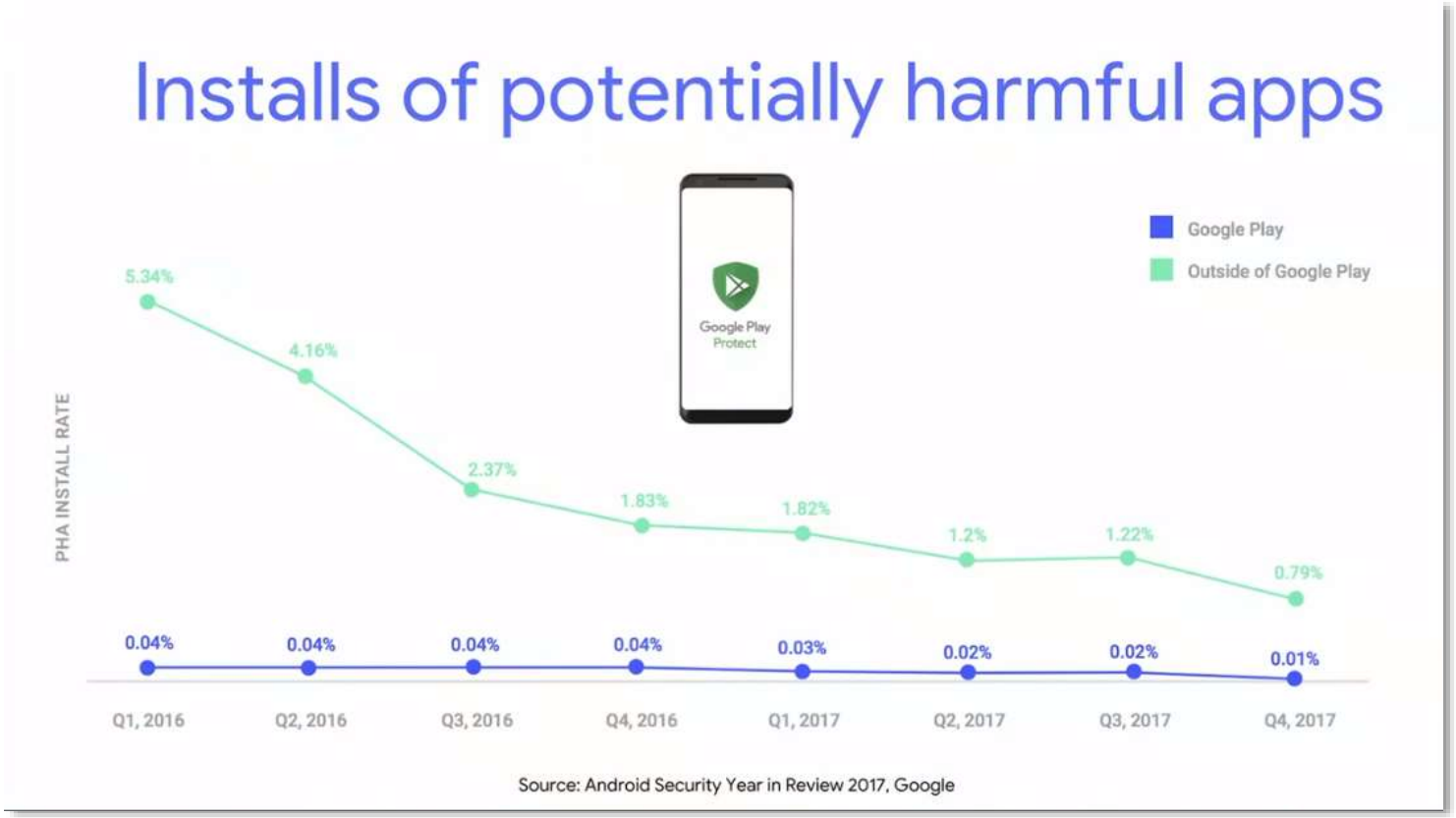
- オープンプラットフォームはセキュリティに貢献する
- 防御側は、数千人のGoole社員＋アーム＋インテル＋クアルコム＋ブロードコム＋Linuxコミュニティ＋学術研究コミュニティ

クローズドプラットフォーム

- 防御側は１社の従業員

2. PHAインストール率

AndroidOSが安全である理由2



PHA（Potentially Harmful Applications）：
**　　マルウエアまたは潜在的に有害なアプリケーション**

緑：Google Play以外からのPHAインストール

青：Google PlayからのPHAインストール

PHAアプリのインストール率

GooglePlayからPHAアプリがインストールされた割合

- 0.2から0.1に減っている２倍減っていることになる。（2017Q3~Q4)
  （グラフをもっと減っているように見えるように加工すればいいのに）

Google Play以外からのPHAアプリインストール率

- Google Protectがクライアントで動作するようになったので明らかに減少している

機械学習を使ってマルウエアの６０%が検知されている

PHAをインストールする確率は雷にあたるのと同じ確率(ﾟДﾟ)

Tip:Android Developer Blog: 2018/5/24

- Keeping 2 billion Android devices safe with machine learning
- 機械学習してるとか言ってるだけで特に有益な情報が出ているわけではない
- https://android-developers.googleblog.com/2018/05/keeping-2-billion-android-devices-safe.html

Google Play Protect

端末やアプリに不正な動作がないかを定期的に確認するアプリ。

- **https://www.android.com/intl/ja_jp/play-protect/**
- **Google I/O 2017発表（2017/7/22頃提供）**
- **設定→Google→セキュリティ→Google Play Protect→ON**

3．エクスプロイトコード価格

AndroidOSが安全である理由3

Exploit（エクスプロイト）コード

- 脆弱性検証するための実証コード
- 脆弱性を利用した悪意ある行為のために書かれたコード

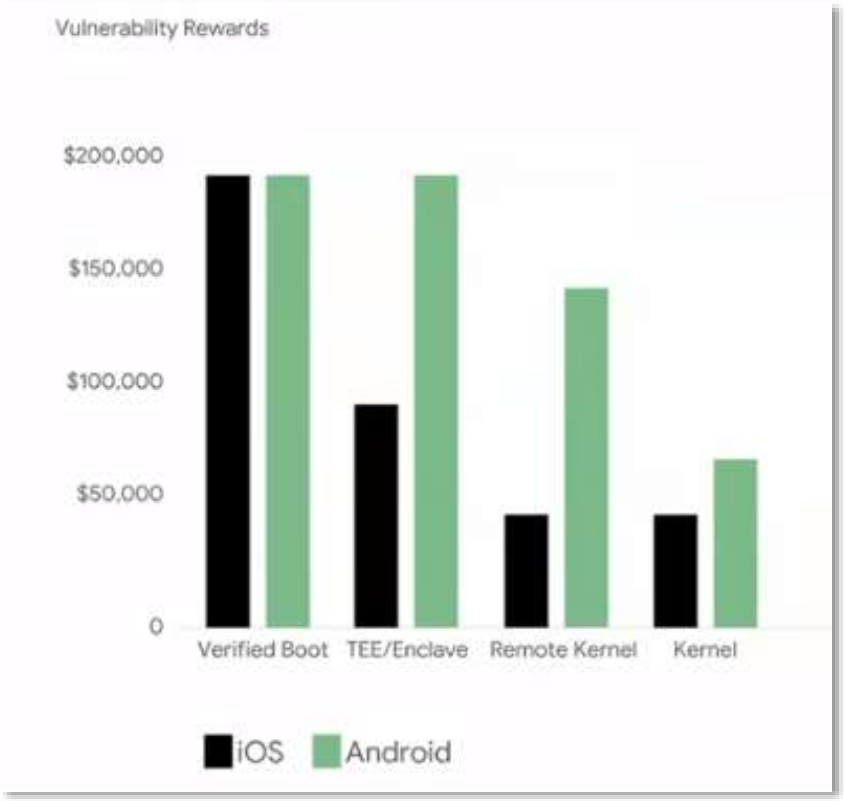
エクスプロイトコード価格

- 売買されている。需要と供給
- 攻撃性が高いものほど高価
- 発見が難しいほど高価

※Androidのエクスプロイトコードは高価→安全であるとGoogleは言っている。

- 言いたいことはわかるけど、ちょっと強引かなと言う気もする。

企業のバグ発見報酬プログラム



緑：Android

黒：iOS

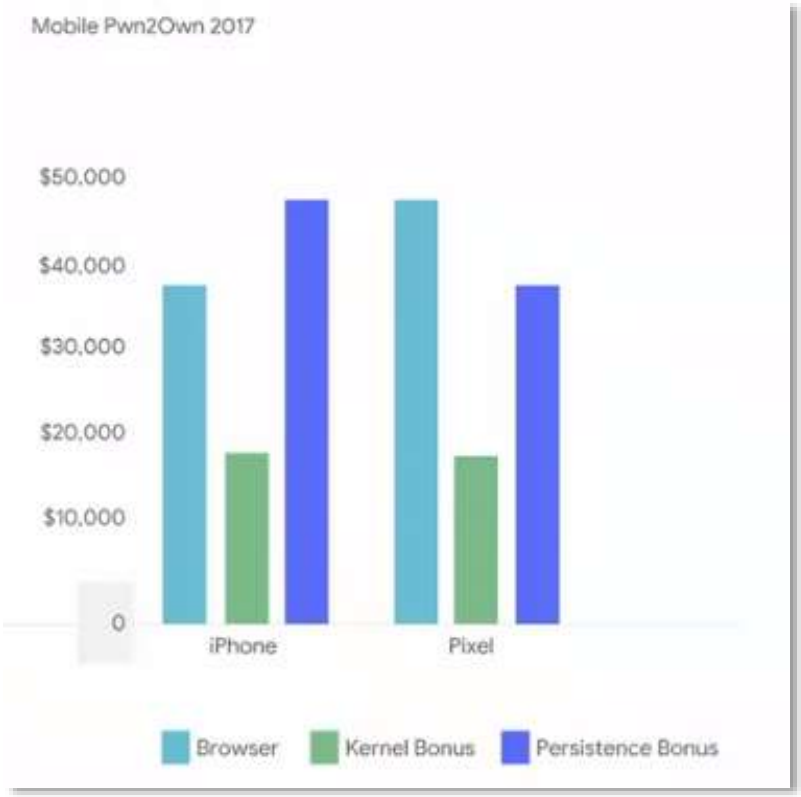
Androidの方が高額を支払っている

- Verified Boot
- TEE/Enclave
- Remote Kernel
- Kernel

Android Security Rewards

- セキュリティバグを報告すると賞金が貰える
- https://www.google.com/about/appsecurity/android-rewards/

ハッキングコンテスト



Mobile Pwn2Own 2017(賞金付き脆弱性発見コンテスト)

AndroidとiPhoneはほぼ同額

アンダーグラウンドマーケット

アンダーグラウンドマーケットでの価格も、他のプラットフォームと同じかそれ以上の価格が観測されている。

よって安全である！

1．Transparency

2．PHAをインストールした発生率

3．エクスプロイトコード価格

Androidのセキュリティは他のモバイルプラットフォームと同じレベルにある。by Google

安全なOSとはなにか？

様々なOSで脆弱性が報告されている。現在ではどのOSが危険、安全という段階ではなく、OSにきちんとセキュリティアップデートが行われているかという事が重要になっている。

- **Android**
- **iOS**
- **Windows**
- **Linux**

セキュリティパッチOEM義務付け

毎月セキュリティパッチを出しているが、OEM端末等は総てが適用されるわけではない

Androidの定例セキュリティパッチをOEMに義務付け

今後は多くの端末でセキュリティパッチが適用される事となる

Project Treble

- デバイスベンダーがOSを効率的にアップデートできるようにする仕組み（Android Oから）
- https://source.android.com/devices/architecture/treble

Androidを安全に使う方法

- OSは最新の状態にアップデートする
- アプリは常に最新のバージョンにアップデートする。
- 信用あるマーケットサイト以外からインストールしない
- アプリのアクセス許可を丁寧に確認する
- 端末のロックを有効にする

サードパーティアプリの脆弱性

IPAテクニカルウオッチ「Android アプリの脆弱性」

IPAに届け出られるAndroidアプリの脆弱性関連情報が2011年後半から増加していることを踏まえ、それらを分析して脆弱性を作り込みやすいポイントをまとめたもの。

20ページと読みやすい

簡易チェックリスト付

IPA簡易チェックリスト

- SDカードに機微な情報を保存しない
- 必要に応じてSDカードに保存するデータを暗号化する。
- ファイル作成時に、ファイルアクセス許可を適切にする。
- アクティビティやサービスに対してアクセス制限をかける
- コンテントプロパイダーに適切なアクセス制限をかける
- デバッグログに機微な情報を含めない
- インテントを送信する際のパラメータに機微な情報を含めない
- 必要以上に権限を要求しない

IPA簡易チェックリスト（現在）

- SDカードに機微な情報を保存しない→一部対応
- 必要に応じてSDカードに保存するデータを暗号化する。 →一部対応
- ファイル作成時に、ファイルアクセス許可を適切にする。 →対応
- アクティビティやサービスに対してアクセス制限をかける→対応
- コンテントプロバイダーに適切なアクセス制限をかける→対応
- デバッグログに機微な情報を含めない→対応
- インテントを送信する際のパラメータに機微な情報を含めない→対応
- 必要以上に権限を要求しない→対応

Androidアプリの脆弱性は減ってきている

OSのバージョンアップによりAndroidアプリの脆弱性は減ってきている

スマホ創世記

- **従来アプリを作らない会社がアプリをリリース**
- **初めてスマフォアプリを作る会社が多かった**
- **特に通信（クライアント側）は経験が少ない**

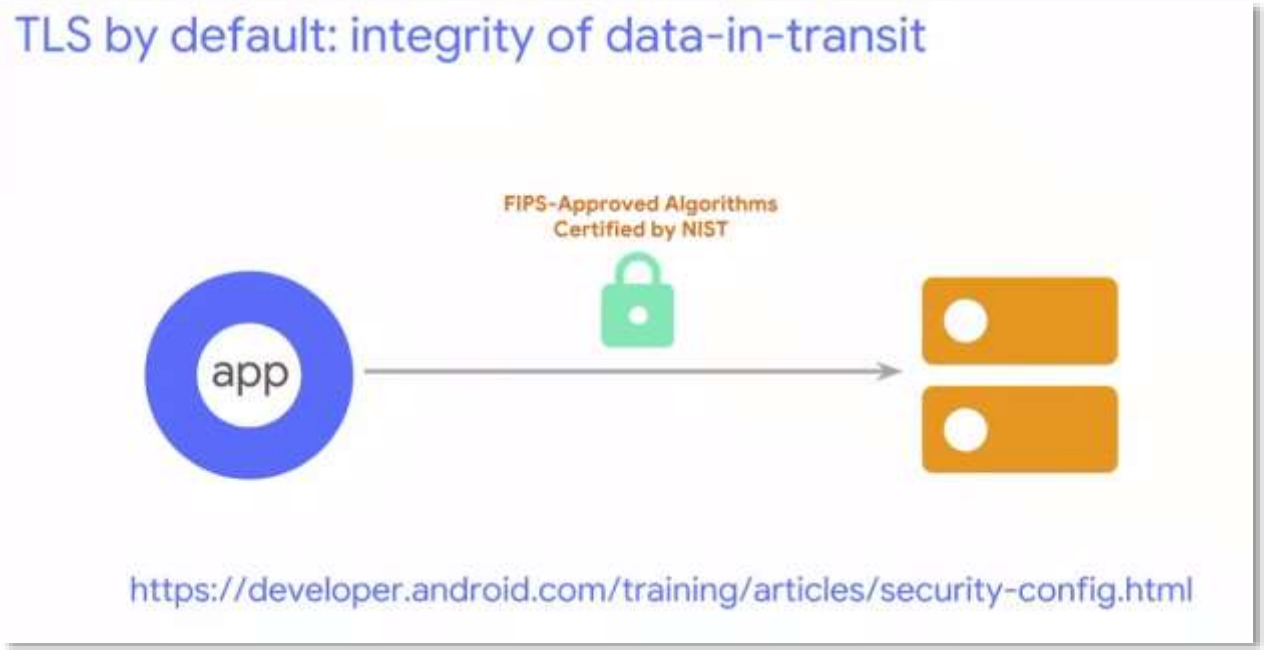
現在

- **OSのバージョンアップによる、脆弱性のあるアプリの減少**
- **Androidアプリ開発者の経験値が上がった。**

Android P

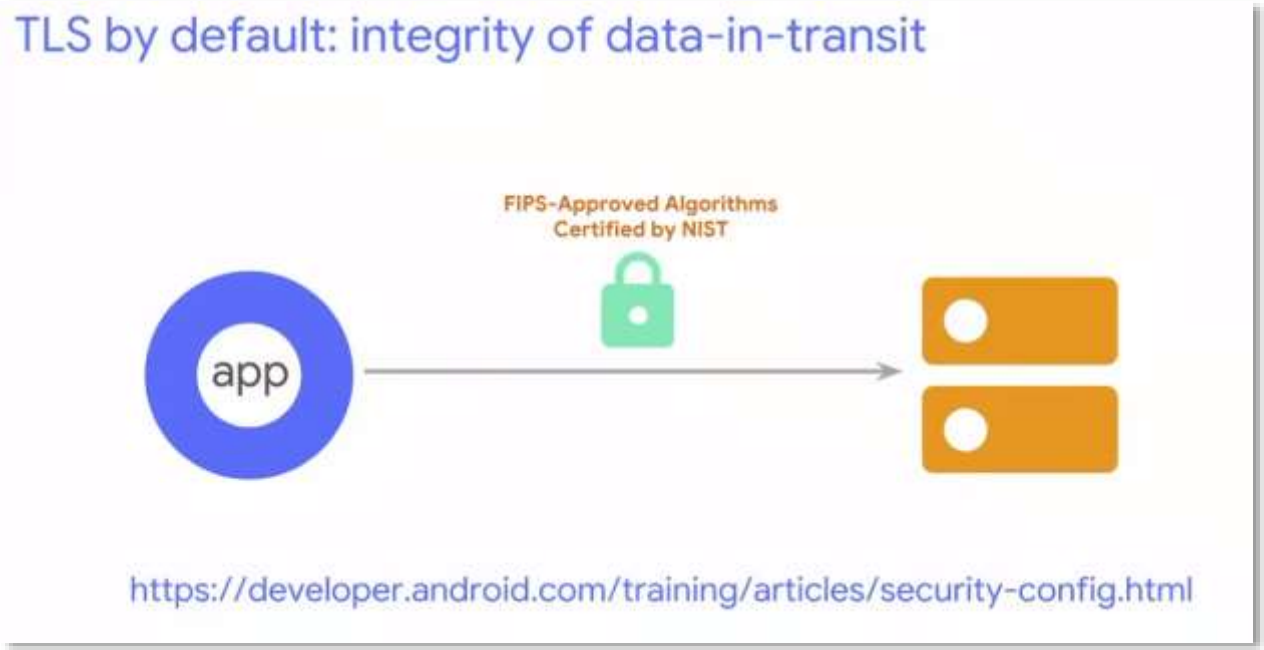
TLS by Default

TLS by Default



- TargtSDKがP以上の時Network Security Configurationファイルで指定するuseCleartextTrafficのデフォルトがtrueからfalseに変更
- 平文通信を禁止するので、通信をしているライブラリを利用している場合は対応済か確認が必要

FIPS対応



- FIPS（フィップス）：Federal Information Procession Standard)アメリカ政府が取り扱う情報機器や情報技術に求められる基準
- Boring SSLがNISTからCAVP署名を受け取ったのでFIPS準拠となった。

StrongBox

About Android KeyStore

Android KeyStoreとは、暗号で使用する鍵を安全に保存する場所及び機能

鍵は暗号化された上でKeyStoreに格納されるため、Root端末であっても鍵を取り出す事はできない。

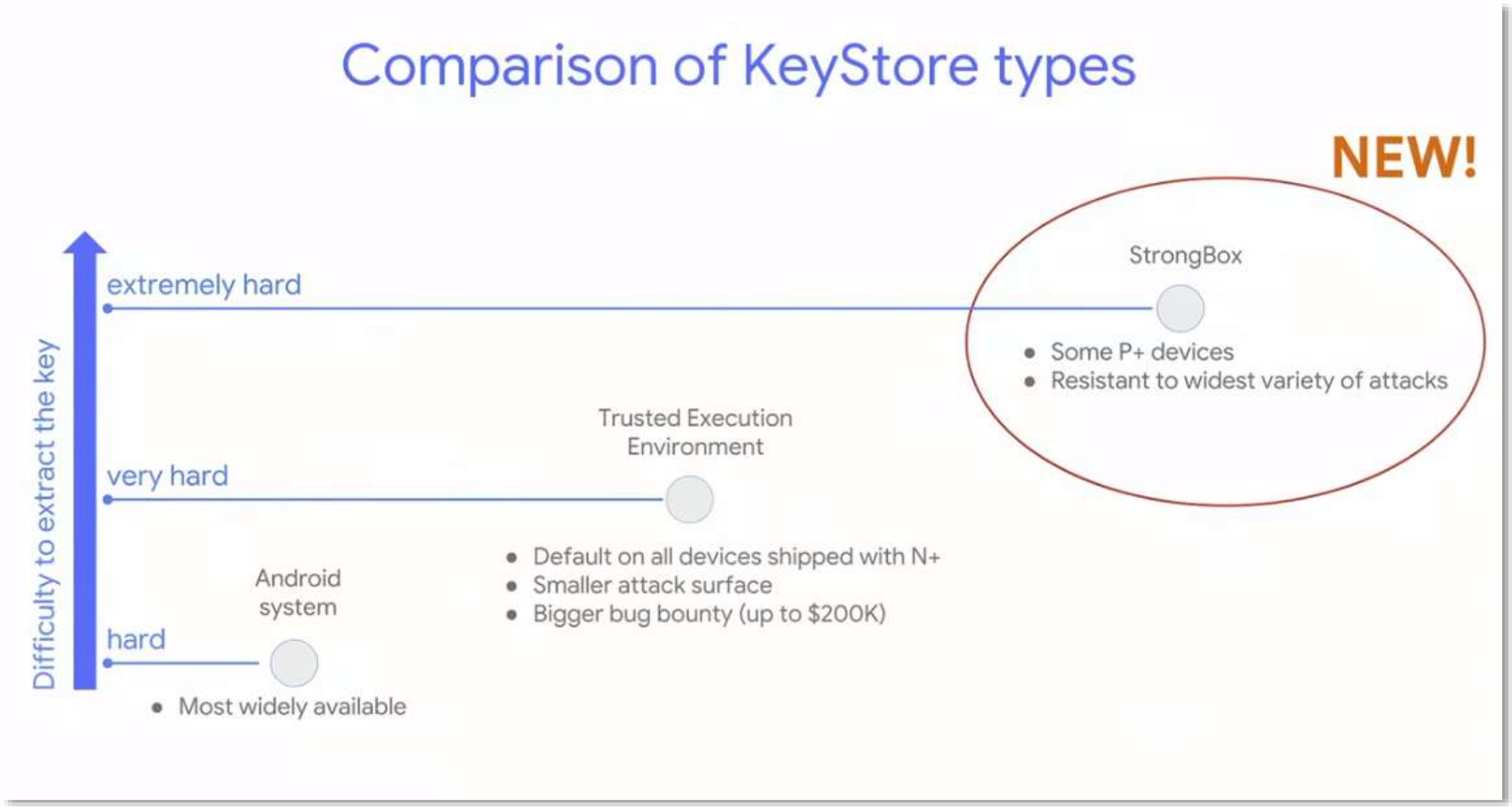
Android4.3

- 導入されたが遅いかつ、鍵の生存期間が不明瞭だったため利用に耐えれなかった

Android6.0　リニューアル

- OSに指紋認証機能追加
- ユーザのロック画面知識ファクタ（Lock Screen Knowledge Factor :LSKF)とAndroid　KeyStoreの関連機能追加
- AES（共通鍵暗号）が利用可能
- 一般的な利用に耐えられるようになった。

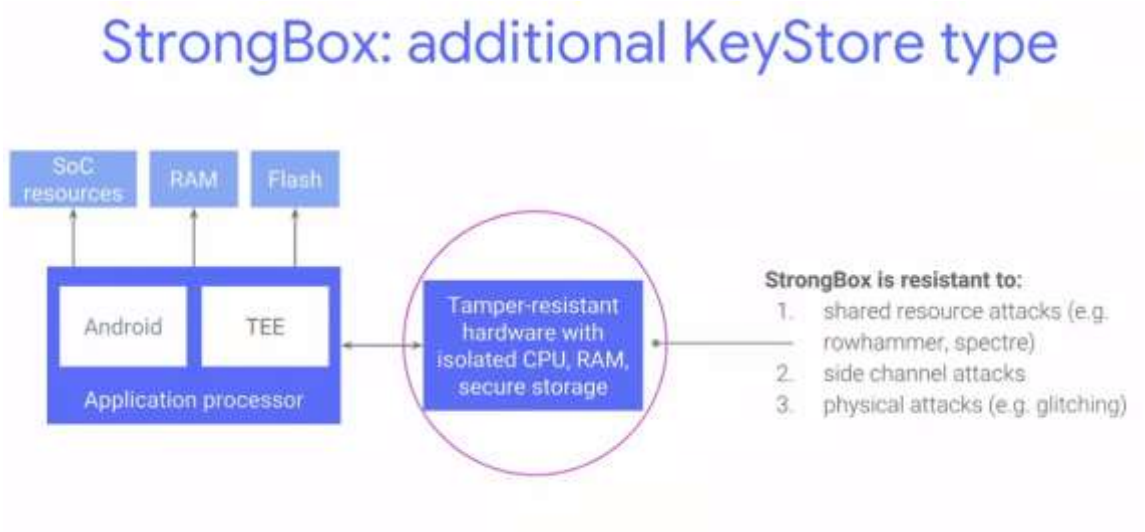
AndroidKeyStoreタイプ



StrongBox:OSはP以上で対応端末のみ利用可能

3つのタイプのキーストア

1. Android System
   - OSがサポート（ソフトウエア）
2. TEE(Trusted Execution Environent)
   - N以上で総ての端末がサポート
3. Strong Box (New!)
   - P以上の端末でサポート（必須ではない）
   - 改ざんに強いハードウエアで実現
   - hasSystemFeature(FEATURE_STRONGBOX_KEYSTORE)



https://developer.android.com/preview/features/security#hardware-security-module

Android P がインストールされたサポート対象の端末では、ハードウェアのセキュリティ モジュール内にある Keymaster HAL の実装である *StrongBox Keymaster* を使用できます。 このモジュールには、独自の CPU、安全なストレージ、真性乱数ジェネレータのほか、パッケージの改ざんやアプリの不正なサイドローディングを防ぐ追加のメカニズムが含まれています。 システムでは、StrongBox Keymaster に格納されている鍵をチェックするときに、信頼できる実行環境（TEE）で鍵の整合性を保証します。

StrongBoxの利用するソースコード

```
// Set StrongBox when generating the key

val kpg = KeyPairGenerator.getInstance(
        KeyProperties.KEY_ALGORITHM_EC, "AndroidKeyStore")

kpg.initialize(KeyGenParameterSpec.Builder(
        alias,
        KeyProperties.PURPOSE_SIGN or KeyProperties.PURPOSE_VERIFY)
        .setDigests(KeyProperties.DIGEST_SHA512)
        .setIsStrongBoxBacked(true)
        .build())

val kp = kpg.generateKeyPair()
```

Use StrongBox

setIsStrongBoxBacked(true)を設定

端末がStrongBox非対応の場合はStrongBoxUnavailableExceptionとなる

Keyguard-bound keys

Keyguard-bound keys

端末がロックされていない状態では復号化できなくする。

ロック画面のライフサイクルに関連付けられる

- **暗号化はいつでもできる**
- **復号化はアンロックした時のみ可能**
- **端末を紛失したり盗まれた時に有効**

Keyguard-bound keys　ソース

```
// Using keyguard-bound keys

val kg = KeyGenerator.getInstance(KEY_ALGORITHM_AES, ...)
kg.init(
  KeyGenParameterSpec.Builder("k", PURPOSE_ENCRYPT or PURPOSE_DECRYPT)
   .setBlockModes(BLOCK_MODE_GCM)
   .setEncryptionPaddings(ENCRYPTION_PADDING_NONE)
   .setUnlockedDeviceRequired(true)
   .build())
val key = kg.generateKey()
...
cipher.init(Cipher.ENCRYPT_MODE, key)
...
cipher.init(Cipher.DECRYPT_MODE, key)
```

The key can only be used to decrypt when the device is unlocked

Initialize a cipher to encrypt sensitive data

Initialize a cipher to decrypt sensitive data when unlocked

setUnlockedDeviceRequired (true)を設定

Lock Down mode

Lock Down Mode

一時的にデバイスにアクセスできなくするモード

セキュリティゲートで端末を人に渡したりする時、警察に端末を押収された、盗難等レアケースではあるが、生体認証、他によって端末にアクセス可能にできるケースがある。（就寝時にも有効かもしれない）

- パスワード,PIN,パターン以外使用できなくなる。
- Smart Lock(身に着けている物、場所、顔認識、音声認識等でロックを外す機能）が無効
- 指紋認証（生体認証）が無効

Lock Down Mode動作

1. 「設定」→「セキュリティと現在地情報」→「ロック画面の設定」→「ロックダウンオプションの表示」
2. 電源長押しでメニューを出して、ロックダウンを入力でロックダウンモード

まとめ

AndroidOS危険危険と言うほどではなくなっています。

AndroidPのセキュリティ変更点については、大きなものはありません。大体細かい所です。

日本アンドロイドの会イベント

2018・10・13日（土曜）

場所：川崎

https://abc.android-group.jp/2018a/

タオソフトウェア株式会社　谷口岳

ありがとうございました。