ANALISA KEAMANAN APLIKASI MOBILE E-COMMERCE BERBASIS ANDROID MENGGUNAKAN MOBILE SECURITY FRAMEWORK

Cholis Hanifurohman[1], Deanna DurbinHutagalung[2],
Dosen Universitas Pamulang
dosen01825@unpam.ac.id

ABSTRAK

Perkembangan *smartphone* di Indonesia yang sedemikian pesat tidak dimungkiri memberikan dampak positif ke beberapa sektor bisnis seperti jual beli *online* dan juga memicu munculnya beragam aplikasi *mobile* khususnya pada *platform android*. Para pelaku *e-commerce* banyak yang memberikan layanan aplikasi *mobile* yang menyesuaikan kebiasaan pengguna saat ini. Dengan jumlah transaksi yang tinggi dilakukan di seluler setiap hari, para pengguna layanan aplikasi mobile *e-commerce* perlu menyadari akan keamanan khususnya yang berkaitan dengan data-data yang sensitif yang tersimpan di dalam *smartphone*. Oleh karena itu perlu dilakukan analisa keamanan terhadap aplikasi dengan melakukan pengujian/pengukuran terhadap tingkat keamanan aplikasi. *Mobile security framework* (MobSF) adalah salah satu metode yang dapat digunakan untuk melakukan pengukuran terhadap keamanan aplikasi. Hasil analisa keamanan menggunakan MobSF diharapkan dapat memberikan kesadaran terhadap pengguna aplikasi dan memberikan masukan kepada pihak pengembang aplikasi untuk terus meningkatkan aspek keamanan.
Kata kunci *: security, smartphone,* android, *e-commerce,* MobSF

ABSTRACT

The rapid growth of smartphones in Indonesia is undeniably having a positive impact on several business sectors such as buying and selling online and also triggering the emergence of various mobile applications, especially on the Android platform. Many e-commerce players provide mobile application services that adjust the current user habits. With a high number of transactions carried out on cellular every day, users of e-commerce mobile application services need to be aware of security, especially relating to sensitive data stored on smartphones. Therefore it is necessary to analyze the security of the application by testing / measuring the level of application security. Mobile security framework (MobSF) is one method that can be used to measure application security. The results of the security analysis using MobSF are expected to provide awareness of application users and provide input to the application developer to continue to improve the security aspects.
Keyword*: security, smartphone,* android, *e-commerce, MobSF*

PENDAHULUAN

Perkembangan *smartphone* di Indonesia yang sedemikian pesat tidak dimungkiri memberikan dampak positif ke sektor perdagangan elektronik atau *e-commerce*, sehingga sektor ini masih menjadi primadona para investor di 2018. Badan Koordinasi Penanaman Modal (BKPM) mengungkapkan, nilai investasi di sektor *e-commerce* pada 2017 mencapai lebih dari USD 5 miliar. Hal ini menjadikan *e-commerce* sebagai sektor ekonomi yang paling strategis saat ini. Asosiasi Pengusaha Ritel Indonesia (APRINDO) memprediksi akan ada lebih dari 50 gerai ritel yang berhenti beroperasi dan mencoba mengubah format bisnis mereka ke arah *online*[1]. Perubahan pola perilaku belanja ini juga ditunjukkan dengan volume transaksi *e-commerce* yang meningkat. Laporan tahunan yang dikeluarkan *We Are Social* menunjukkan, prosentase masyarakat Indonesia yang membeli barang dan jasa secara *online* dalam kurun waktu sebulan di 2017 mencapai 41% dari total populasi, meningkat 15% dibanding tahun 2016 yang hanya 26%[2]. Pesatnya pertumbuhan *e-commerce* ini juga mendorong aplikasi-aplikasi *mobilee-commerce* berkembang sejalan dengan kebutuhan masyarakat akan layanan berbasis aplikasi mobile yang berjalan pada *smartphone*. Kementerian Komunikasi dan Informatika memprediksi pada 2018 jumlah pengguna *smartphone* aktif di Indonesia mencapai 100 juta orang. Dengan jumlah tersebut, Indonesia akan menjadi negara pengguna *smartphone* terbesar keempat di dunia setelah Tiongkok, India, dan Amerika. Meskipun transaksi *e-Commerce* pada 2016 masih didominasi PC, tapi pada 2017 kondisi akan berubah. Jumlah transaksimelalui *smartphone* diprediksi mengalahkan transaksi dari PC.

Untuk itu, dengan tren perilaku konsumen saat ini, pelaku *e-commerce* harus bisa memaksimalkan aplikasi *mobile* yang dimiliki. Proses tersebut dapat dimulai dengan optimalisasi UI dan UX, termasuk promo eksklusif bagi pengguna aplikasi.

E-commerce telah membuat hidup lebih mudah bagi banyak orangorang di seluruh dunia sehingga melakukan transaksi harian yang dilakukan secara nirkabel dengan nyaman tetapi juga menimbulkan beberapa ancaman keamanan[3].

Celah-celah keamanan yang terdapat di aplikasi dapat digunakan penyerang untuk mencuri informasi penting di dalam *smartphone*, dimana informasi merupakan salah suatu aset penting dan sangat berharga disajikan dalam berbagai format berupa : catatan, lisan, elektronik, pos, dan audio visual [4].

Oleh karena itu perlu dilakukan analisa keamanan terhadap aplikasi dengan melakukan pengujian/pengukuran terhadap tingkat keamanan aplikasi. *Mobile security framework* (MobSF) adalah salah satu metode yang dapat digunakan untuk melakukan pengukuran terhadap keamanan aplikasi. Hasil analisa keamanan menggunakan MobSF diharapkan dapat memberikan kesadaran terhadap pengguna aplikasi dan memberikan masukan kepada pihak pengembang aplikasi untuk terus meningkatkan aspek keamanan dan dari sisi penguna aplikasi menyadari akan risiko keamanan terhadap setiap aplikasi *mobilee-commerce* yang mereka gunakan.

BAHAN DAN METODE

E-commerce

Mobile e-commerce, didefinisikan sebagai penyampaian *e-commerce* secara langsung ke tangan pelanggan, di mana pun melalui teknologi nirkabel yang pada awalnya diciptakan oleh Kevin Duffey pada tahun 1997. Perdagangan mobile adhoc berlangsung antara beberapa *node* yang dekat satu sama lain tanpa mengandalkan pada layanan infrastruktur apa pun [4].

Dalam seluruh proses transaksi *mobilee-commerce*sistem transaksi, ada tiga faktor utama yang tidak amanyang berasal dari *mobile terminals*, *mobile radio interface*dan*network-side*[5].

Faktor tidak aman *mobile terminals*

Faktor-faktor tidak aman *mobile terminals* terutama dimanifestasikan dalam identitas pengguna, informasi akun, dan kunci otentikasi dan sebagainya. Misalnya, orang lain yang mendapatkan *mobile terminals*pengguna cenderung memalsukan identitas pengguna untuk melakukan beberapa kegiatan ilegal.

Faktor-faktor tidak *mobile radio interface*

Sebagai komunikasi antara terminal seluler dan tetapjaringan dalam transmisi

nirkabel bergantung pada antarmuka nirkabel terbuka untuk mengirimkan, setiap orang yang memiliki perangkat nirkabel yang sesuai akan memiliki kesempatan untuk mendapatkan informasi melalui penyadapannya melalui saluran nirkabel, dan bahkan dapat memodifikasi, menghapus atau mengirim kembali informasi, yang menimbulkan ancaman terhadap aktivitas perdagangan.

Faktor-faktor jaringan yang tidak aman

Jaringan terutama mengacu pada jaringan nirkabel, gateway dan jalur kabel. Jika informasi tidak dilindungi ketika dikirim dalam jaringan nirkabel, jaringan kabel dan dikonversi oleh gateway, kemungkinan akan terekspos menyebabkan ancaman terhadap kegiatan perdagangan.

Ancaman pada kemananan android

Mekanisme berbasis *permission*/izin disediakan untuk keamanan aplikasi android yang mengatur akses aplikasi android pihak ketiga ke sumber daya penting pada perangkat. Mekanisme ini sangat dikritik karena kontrolnya yang kasar terhadap izin aplikasi dan manajemen izin yang tidak efisien, oleh pengembang, dan pengguna. Misalnya, pengguna diizinkan untuk menerima semua permintaan izin dari aplikasi untuk menginstalnya atau menolak instalasi aplikasi. Bagian ini menjelaskan masalah keamanan utama android yang menyebabkan kebocoran informasi pengguna dan menyebabkan hilangnya privasi pengguna[6].

Setidaknya ada 4 ancaman pada kemanan android :

1. Kebocoran Data
   Aplikasi android yang bocor dapat menempatkan informasi yang sensitif bagi pengguna di lokasi yang tidak aman di perangkat atau dapat mengirim informasi identifikasi perangkat, misalnya metadata aplikasi seperti detail jaringan. Lokasi perangkat yang tidak aman ini dapat diakses oleh aplikasi jahat lainnya pada perangkat yang sama. Data atau informasi sensitif yang bocor menyebabkan perangkat menjadi kondisi kritis. Eksploitasi kerentanan ini sangat mudah karena penyerang dapat memperoleh akses ke bagian perangkat tempat data sensitif disimpan. Dampak dari kebocoran data perangkat Android sangat parah. Sesuai dengan situs web berita grup peneliti keamanan, 58% perangkat Android memiliki kebocoran privasi dan sekitar 3% memiliki kebocoran PII (*personally identifiable information*).
2. Eskalasi hak istimewa
   Kekurangan keamanan mekanisme izin android dapat menyebabkan peningkatan eskalasi hak istimewa yang disebabkan oleh aplikasi yang dikompromikan. Para penulis menggambarkan peningkatan hak istimewa sebagai: Aplikasi dengan izin yang lebih sedikit (penelepon yang tidak memiliki hak istimewa) tidak dibatasi untuk mengakses komponen aplikasi yang lebih istimewa (hak istimewa). Contoh eskalasi hak istimewa dapat diberikan sebagai - kejahatan lokal dapat mengeksekusi kode arbitrer di kernel tanpa memiliki hak istimewa untuk melakukannya. Hal ini dapat menyebabkan kompromi total pada sistem operasi yang menyebabkan korupsi pada sistem operasi dan menyelesaikan perbaikan perangkat. Sesuai dengan database*Common vulnerabilities and exposures* (CVE), kerentanan eskalasi hak istimewa yang kritis ditemukan di android versi 6 dan di atasnya. Pelanggaran hak istimewa di android membuat jutaan pengguna berisiko dibajak *smartphone*-nya.
3. Pengemasan ulang aplikasi
   Proses pembongkaran / dekompilasi file .apk menggunakan teknik rekayasa balik (*reverse engineering*) dan menambahkan/menyusupkan kode berbahaya ke dalam kode sumber utama dikenal sebagai pengemasan ulang aplikasi android. Untuk pengguna android, menjadi sulit untuk membedakan antara aplikasi jahat yang dipaket ulang dan aplikasi normal karena aplikasi yang dikemas ulang biasanya berfungsi dengan cara yang sama dengan yang sah.
4. Serangan DDos
   Dalam serangan DDos, penyerang berusaha membuat perangkat atau sumber daya tidak tersedia untuk

penggunaan yang dimaksudkan dengan mengganggu layanan perangkat host untuk sementara atau tidak terbatas. Sesuai dengan Laporan dari *Symantec Internet Security*, sekitar 7,2% aplikasi android mengalami serangan DDos[6].

Android Dangerous permissions

Dangerous permissions mencakup area di mana aplikasi menginginkan data atau sumber daya yang melibatkan informasi pribadi pengguna, atau berpotensi memengaruhi data yang disimpan pengguna atau pengoperasian aplikasi lain. Misalnya, kemampuan membaca kontak pengguna adalah *dangerous permissions*. Jika suatu aplikasi menyatakan bahwa ia membutuhkan*dangerous permissions*, pengguna harus secara eksplisit memberikan izin kepada aplikasi tersebut. Sampai pengguna menyetujui izin, aplikasi tidak dapat menyediakan fungsionalitas yang tergantung pada izin tersebut.

Untuk menggunakan *dangerous permissions*, aplikasi harus meminta pengguna untuk memberikan izin saat *runtime*. Untuk detail lebih lanjut tentang bagaimana pengguna diminta, lihat Permintaan prompt untuk izin berbahaya.[7]

MobSF

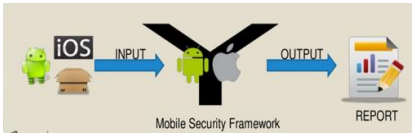
Mobile Security Framework (MobSF) adalah framework yang digunakan untuk pengujian penetreasi terhadap aplikasi seluler (Android / iOS / Windows) otomatis yang mampu melakukan analisis statis, dinamis, dan malware. MobSF dapat digunakan untuk analisis keamanan yang efektif dan cepat dari aplikasi seluler Android, iOS dan Windows dan mendukung kedua binari (APK, IPA & APPX) dan kode sumber zip. MobSF dapat melakukan pengujian aplikasi dinamis saat runtime untuk aplikasi Android dan memiliki kemampuan fuzzing API Web yang didukung oleh CapFuzz, pemindai keamanan khusus Web API. MobSF dirancang untuk membuat integrasi CI / CD atau DevSecOps secara mulus[8].

Analisis Statis

Analisis statis adalah salah satu tahapan pengujian aplikasi seluler. Menurut pentester Hacken, kerangka kerja open source yang paling nyaman adalah MobSF[9].Secara umum analisis statis menggunakan MobS pada aplikasi mobile *e-commerce* mencakup:

- SSL bypass
- Weak Crypto
- Permissions
- Hardcode secrets
- Malware check

Beriku ini alur kerja analisis statis menggunakan MobSF.



Gambar 1. Analisis Statis pada MobSF

Tahapan proses analisis statis menggunakan MobSF :

Langkah 1

Setelah berhasil melakukan instalasi MobSF, jalankan script berikut pada direktori MobSF : *$.run.sh.*

Kemudian pada browser diakses melalui alamat http://127.0.0.1:8000 untuk mendapatkan beberapa fitur seperti :

- file upload
- view previous scan reports
- transition to API documentation
- transition to GitHub project

Gambar 2. Halaman Home pada MobSF

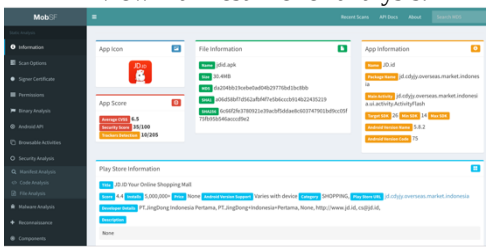
Langkah 2

Setelah file diunduh dan dianalisis, halaman dengan hasil analisis muncul. Ada menu di sebelah kiri yang memungkinkan untuk menavigasi dengan cepat di seluruh halaman (hasilnya cukup banyak). Berikut ini informasi bermanfaat dalam tangkapan layar ini:

- application hash sum;
- Supported Android OS Versions;
- the number and type of components (exported or not): it's important, as exported components can lead to critical vulnerabilities;
- the ability to view and download java- and smali-files that can be

analyzed either by other tools or manually;

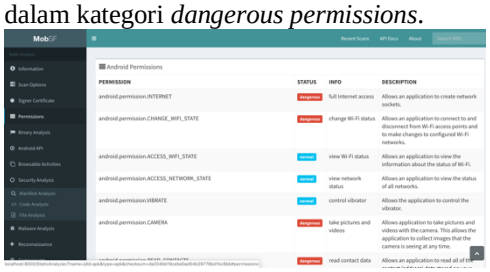
- view manifest file for analysis.



Gambar 3. Halaman Hasil Analisis pada MobSF

Langkah 3

MobSF dapat melihat deskripsi analisis izin, yang terdapat dalam file AndroidManifest.xml. MobSF menganalisis izin aplikasi Android, menentukan statusnya terkait kekritisan, dan deskripsi izin. Di sini perlu memahami arsitektur OS Android untuk menilai tingkat kekritisannya yang sebenarnya. Dari informasi izin ini akan dapat terlihat seberapa banyak izin yang diminta aplikasi dan seberapa banyak yang dalam kategori *dangerous permissions*.



Gambar 3. Halaman Hasil Analisis izin pada MobSF

Selain analisis izin, pada bagian Security Analysis akan ditampilkan juga hasil dari analisis source code dan analisis file yang merupakan aset dari aplikasi serta analisis malware dari domain-domain yang terdapat dalam aplikasi tersebut.

Dari hasil analisis statis aplikasi dan kode sumber memberikan pemahaman dasar tentang arsitektur aplikasi android dan beberapa serangan yang potensial. Analisis statis digunakan sebagai awal dari setiap pentesting aplikasi.

Untuk lebih lengkapnya analisis statis dilanjutkan dengan analisis dinamis dimana MobSF juga menyediakan analisis dinamis tersebut.

Aplikasi Mobile E-commrec di Indonesia

Peta penggunaan e-commerce secara global maupun di indonesia dapat diketahui melalui laporan berkala yang di keluarkan oleh iPrice.

iPrice Group adalah situs meta-search yang beroperasi di Indonesia dan enam negara lain di Asia Tenggara, yakni; Malaysia, Singapura, Filipina, Thailand, Vietnam dan Hong Kong. iPrice bermitra dengan sejumlah brand terbesar di kawasan ini, seperti Tokopedia, Bukalapak, Lazada, Shopee, Zalora, Gojek, Traveloka, Klook dan banyak lagi. Secara berkala, iPrice Group juga merilis laporan mendalam mengenai *e-commerce*, startup dan topik terkait lainnya[10].

Adapun 5 besar aplikasi mobile e-commerce yang berbasis android pada semester kedua tahun 2019 (ditampilkan dalam bentuk singkatan nama) adalah sebagai berikut :

Tabel 1 Aplikasi mobile *e-commerce*

Peringkat	Aplikasi	Versi
1	SP	v2.41.06
2	TP	v.3.35
3	LZ	v.6.32.0
4	BL	v.4.42.5
5	SR	v.0.3.8

HASIL

Analisis yang dilakukan untuk analisis keamanan aplikasi mobile *e-commerce* adalah analisis statis dengan melihat seluruh aspek analisis statis dalam MobSF. android Berdasarkan hasil analsis statis yang dilakukan didapatkan hasil seperti yang ditampilkan dalam tabel berikut ini :

Tabel 2 Hasil analisis statis

App	Weak Crypto	SSL Bypass	Dangerous Permissions	Hardcode Secret	Root Detection	Domain Malware Check
SP	NO	YES	YES	NO	YES	GOOD
TP	NO	YES	YES	YES	YES	GOOD
LZ	YES	YES	YES	YES	YES	GOOD
BL	YES	YES	YES	NO	YES	GOOD
SR	NO	YES	YES	NO	YES	GOOD

Weak Crypto

Analisis weak crypto dilakukan dengan melihat apakah terdapat impelemntasi algoritma kriptografi yang lemah atau penggunaan lagoritma kriptografi yang sudah usang atau sudah dianggap tidak layak. Dari ke-5 aplikasi yang dilakukan analisis, terdapat 2 aplikasi yaitu LZ dan BL yang masih menggunakan algoritma SHA-1 sebagai algoritma yang digunakan dalam sertifikanya digitalnya untuk code signing. Hasil penelitian menunjukkan bahwa penggunaan SHA-1 untuk sertifikat atau untuk otentikasi *handshake* di TLS, SSH

atau IKE berbahaya, karena sudah terbukti dapat dilakukan *collision attack*dan bisa disalahgunakan oleh penyerang. SHA-1 telah rusak sejak 2004, tetapi masih digunakan di banyak sistem keamanan; sehinggadisarankan kepada pengguna untuk menghapus dukungan SHA-1 untuk menghindari serangan[11].

SSL *bypass*

Analisis SSL *bypass* dilakukan dengan melakukan check apakah terdapat service yang melibatkan protokol http yang tidak mewajibkan penggunakan SSL sebagai persyaratakan keamanan transaksi dalam protokol http menggunakan SSLseperti mengijinkan http di manifest, atauterdapat string berkonten http:// yang merupakan *weak implementation*.Ketika mengimplementasikan ke sebuah aplikasi seharusnyamenggunakan komunikasi https baik digunakan secara *native*(contoh : web service), atau dalam yang digunakan dalam*webview*yang mengakes sebuah webpage melelui webview atau mengimplementasikan keduanya. Hasil analisis terhadap SSL *bypass* menunjukkan kelima aplikasi terdapat implementasi yang memungkinkan terjadinya SSL *bypass*.

Dangerous permissions

Analisis terhadap *permission* lebih diarahkan pada seberapa banyak *dangerous permissions*yang digunakan oleh aplikasi. Semua aplikasi yang dilakukan analisis menunjukkan bahwa semuanya mengandung *dangerous permissions*. Hal ini menunjukkan bahwa aplikasi tersebut memungkinkan mengambil data-data pribadi dari perangkat jika pengguna memberikan hak izin.

Hardcode secret

Analisis *hardcode secret* dilakukan dengan melakukan pengecekan apakah terdapat *credential, password, key* atau informasi rahasia lainnya yang tersimpan secara *hardcode* di dalam aplikasi. Dari kelima aplikasi terdapat 2 aplikasi yang kemungkinan terdapat *hardcode secret* yaitu TP dan LZ yang ditemukan dalam bentuk file. File yang ditemukan diduga menyimpan data

Root Detection

Analisis *root detection* dilakukan dengan melakukan check apakah aplikasi mempunyai fungsi untuk melakukan deteksi akses *root* terhadap perangkat android yang digunakan. Dimana akses memungkinkan untuk akses langsung ke dalam sistem termasuk data-data yang dimiliki aplikasi. Hasil analisis *root detection* menunjukkan kelima aplikasi sudah menyediakan mekanisme *root detection* di dalamnya, sehingga aplikasi tersebut tidak akan berjalan pada perangkat yang sudah terdapat akses root (*rooted device*) atau minimal memberikan informasi kepada pengguna bahwa perangkat yang digunakan dalam keadaan akses *root* dan berpotensi membahayakan.

Domain Malware check

Analisis *domain malware check* dilakukan dengan melakukan check apakah domain-domain yang terdapat dalam aplikasi terindikasi dalam kategori domain yang mengandung *malware* atau tidak. Hasil analisis menunjukkan bahwa kelima aplikasi tidak menunjukan bahwa terdapat domain yang terindikasi malware.

KESIMPULAN

Hasil analisis keamanan yang dilakukan pada aplikasi *mobile e-commerce* yaitu SP, TP, LZ, BL dan SR yang merupakan lima besar *mobile e-commerce* berbasis android paling populer di Indonesia menunjukkan bahwa beberapa celah keamanan masih terdapat dari di kelima aplikasi tersebut. Celah-celah kemanan yang ditemukan dapat menjadi security awareness untuk pengguna aplikasi tersebut yang jumlahlah cukup banyak di Indonesia.Untuk pengembang aplikasi android khususnya untuk *e-commerce*hasil analisis ini dapat dijadikan catatan keamanan yang harus diperbaiki.

Analisis yang dilakukan hanya menggunakan analisis statis yang merupakan tahapan awal dalam analisis keamanan sehingga perlu dilanjutkan menggunakan anlisis dinamis untuk mendapatkan hasil yang lebih lengkap analisis keamanan yang dilakukan.

DAFTAR PUSTAKA

[1] https://swa.co.id/swa/listed-articles/prediksi-tren-e-commerce-indonesia-2018 diakses pada tanggal 14 Oktober 2019.

[2] https://wearesocial.com/special-reports/digital-southeast-asia-2017 diakses pada tanggal 14 Oktober 2019.

[3] Usman Jibril Wushishi, Akintoye Oluwasegun Ogundiya, "Mobile Commerce And Security Issues" in International Journal of Scientific Research Engineering & Technology (IJSRET), ISSN 2278 – 0882 Vol 3, Issue 4, July 2014.

[4] Keith Makan & Scott Alexander-Brown, "Android Security Cookbok", Pack Publishing, 2013.

[5] Ali Mirarab, AbdolReza Rasouli kenari, "Study of secure m-commerce, challenges and solutions" in ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 2, No.8 , March 2014

[6] Persin Kaur Granthi, Mrs. S. M. Bansode, "Android Security: A Survey of Security Issues And Defenses", in International Research Journal of Engineering and Technology (IRJET), Vol. 4Issue: 07, July 2017

[7] https://developer.android.com/ training/permissions/requesting#normal-dangerous diakses pada tanggal 14 Oktober 2019.

[8] https://github.com/MobSF/Mobile-Security-Framework-MobSFdiakses pada tanggal 14 Oktober 2019.

[9]https://hacken.io/research/industry-news-and-insights/static-analysis-of-android-mobile-applications-mobsf-manual/diakses pada tanggal 14 Oktober 2019.

[10] https://iprice.co.id/insights/mapofecommerce/diakses pada tanggal 14 Oktober 2019.

[11] Gaetan Leurent and Thomas Peyrin,"From Collisions to Chosen-Prefix Collisions Application to Full SHA-1", in book: Advances in Cryptology – EUROCRYPT 2019, pp.527-555