promon

Major Government Apps In Asia Leak Sensitive Data And Lack Basic Security

~ EGOV APPS ARE ON THE RISE - BUT HOW SECURE ARE THEY?

Mobile eGovernment (eGov) apps create a unique opportunity for governments to interact with their citizens and provide streamlined solutions for them - from eID's and healthcare apps to tax services apps. eGovernment apps cover a wide range of services and can be highly beneficial for a country's citizens.

But these apps can also contain a great deal of sensitive information that needs to be kept safe. When it comes to protecting user data, governments should lead by example and set the standard. If not implementing security mechanisms to protect against common attack methods, these apps ultimately put citizens' data at risk.

Due to COVID-19, governments have accelerated the digitization of their citizen interactions by several years. And because of the rise in popularity of eGov apps in Asia and their rapidly increasing number of users, we wanted to analyse the top apps in this sector to assess for any major vulnerabilities and weak spots in the overall eGov app landscape.

Our mission was to find out if the apps have strong enough security mechanisms in place or if they contain vulnerabilities that could potentially jeopardise citizens' data.

When governments fail to implement proper security for their apps, it opens up the app to be easily manipulated by malware or reverse-engineered by bad actors, potentially leading to account takeovers, data leakage, and fraud.

~ METHODOLOGY

As part of our analysis, we assessed 12 of the top Android and iOS mobile eGov apps in the Asia-Pacific (APAC) region. We believe that the selected apps provide a window into the security flaws most popular eGov apps in the APAC region contain.

Our chosen apps provide citizens with services such as access to health information including, electronic healthcare records, COVID-19 test results, and other personal digital services.

It is important to note that this research is not a comprehensive study. Our researchers conducted a security assessment using free and easily accessible tools. Still, we found vulnerabilities in most apps that raise concern, and further analysis could reveal even more weaknesses.

We conducted both static and dynamic analysis, as well as assessing whether any runtime security and anti-malware capabilities were in place.

Static analysis: In these types of attacks, malicious users attempt to decompile or disassemble the apps offline on a local device. During a static attack, an attacker may look at the app code and attempt to reverse engineer it to understand how the app functions. By doing this, they may find security vulnerabilities within the app or sensitive information to steal.

Runtime analysis: At runtime, the attacker can employ a variety of tools and techniques to analyse or modify the app. It is easier than ever before for an attacker to deploy various techniques like jailbreaking, rooting, hooking, and more in order to, for example, steal the app's decryption keys, intercept communication to servers, and more.

~ THE WORRYING RESULTS

Governments should be the ones to set an example when it comes to the protection of privacy and user data, but our results show that they - more often than not - don't.

While some had varying levels of protection, our analysis found that the majority of the eGov apps had few security mechanisms in place, and some were not even protected at all.

About 60% of the tested apps leak sensitive data

In our assessment, we wanted to find out if the eGov apps were protecting app assets, such as cloud storage keys, API keys, certificates and tokens, or if these could be easily located in the source code of the apps. If such secrets are exposed, it would indicate that the app is more prone to serious attacks and data leakage.

Our researchers found a large number of secrets exposed in the app code of many of the apps. Overall, they found that about 60% of the eGov apps did not protect sensitive app data.

The findings outlined below are only those we consider as the most critical, although other findings could be included here as well.

~ APPS EXPOSING USER DATA

Typically, eGov apps are designed to track sensitive data, including personally identifiable information (PII). This data is often cached before being uploaded to official channels for tracking purposes etc.

It was found that it can be possible to scrape this data from a device. In some cases this data is conveniently stored in well-formatted, but more importantly, unencrypted, SQL databases showing when and where a user had been located.

Even in cases where PII is stored in an encrypted form, because of the lack of security, the storage mediums can be reverse-engineered. Encryption keys can be extracted easily through hooking techniques and, in some cases, were even present in the app code base itself.

The reputational damage resulting from a data breach can be devastating. News travels fast, and organisations can become a global news story within a matter of hours of a breach being disclosed. Ultimately this propagates a loss in consumer trust and can cause irreparable damage to the parties involved.

With GDPR regulations beginning to permeate across the world, a very large microscope has formed to govern the implementation of data privacy laws and large organizational fines for data breaches.

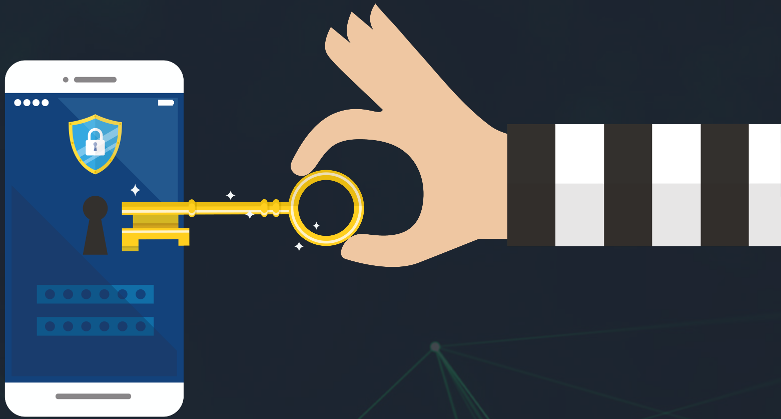
~ APPS LEAKING API KEYS AND CERTIFICATES

In almost all cases, it was possible to obtain certificates used to perform Secure Sockets Layer(SSL) and Transport Layer Security(TLS) pinning. Moreover, amongst those with easily identifiable certificates, the worst offending cases contained root certificate authority certificates embedded into the app.

Moreover, many of the apps under test included API keys, including Amazon Web Services(AWS) and JSON Web Encryption tokens(JWE).

Leaking public CA certificates in itself may seem low risk. The certificates are public in nature and are used for data encryption and handshaking between client and server through the use of asymmetric cryptography. However, in many apps, the endpoint API's can be easily reverse-engineered and exploited by malicious software. Valid API keys and certificates can be obtained when not effectively protected and provides the ability of malicious software to be undetected by the endpoints they communicate with.

This can lead to a multitude of attack vectors, including impersonation of official channels, scraping of PII, distributed denial of service attacks resulting in operational downtime and even database pollution.



~ SENIOR TECHNICAL DIRECTOR AT PROMON, ANDREW WHALEY COMMENTS:

The level of vulnerability of these government apps isn't surprising and is similar to what we see across the board. Interestingly, some of these apps are supposed to monitor user compliance with local lockdown measures. Therefore there is a real incentive for users to exploit these vulnerabilities.

The lack of integrity controls or secure storage of certificates and API keys would mean that it's relatively easy to modify the app to report that a user is at home observing quarantine measures when in fact, they are out at a nightclub! Securing apps using suitable tools for iOS and Android would make it extremely difficult for somebody to bypass these controls. Therefore it's surprising that it hasn't been done in these cases.

~ AN 'ILLUSION OF SECURITY'

Very few of the apps under test made use of in-app security controls, of those that did, the majority utilised open-source implementations to provide what can only be described as an 'illusion of security'.

These implementations provide an unfounded sense of comfort that an app is safe from attack. Open-source security software solutions are inherently vulnerable to exploits and are easily identifiable, often have freely available tools to automatically circumvent them, and can ultimately be removed from any app, leaving them vulnerable to reverse engineering and exploitation.

It was found on iOS platforms that even fewer of the apps under test implemented security. It is a common misconception that iOS is a more secure operating system than other platforms. This belief ultimately originates from iOS being a closed source operating system and therefore at less risk of being exploited. Recent jailbreaking vulnerabilities, however, paint a very different picture and show that iOS is no less vulnerable than Android or any other operating system.

Even a well-secured app that does not easily reveal secrets on one platform can be easily exploited through counterpart apps running on alternative platforms. Analysing equivalent apps on iOS, where well-designed security was not being incorporated, easily revealed secrets that were otherwise difficult to obtain from other sources.

Security sensitive apps need to incorporate countermeasures wherever they are being executed rather than simply relying on operating system vendors 'to do things right'.

~ THE MAJORITY OF THE TESTED APPS LACK BASIC SECURITY

Mobile apps are uniquely exposed to cyber attacks since the app code, by its very nature, has to be released out into the wild.
For this reason, it is imperative that apps provided by governments should meet certain security standards and at least have basic security mechanisms in place, such as those recommended by OWASP.

50% of the apps analysed do not use code obfuscation

- which is considered a basic method for protection. Code obfuscation makes the source code of an app challenging to read and comprehend. It is a technique used to prevent attackers from reverse engineering an app's code and can also do the job of, for example, developing targeted malware more time-consuming for an attacker. By not using code obfuscation, the app code is exposed to malicious actors.

More than 65% of the tested apps are not detecting if an attacker is analysing the app at runtime,

using basic and widely used analytic tools. Attacks against mobile apps often start by using an emulator for the mobile operating system where the targeted app will be run and analysed. Whatever an attacker wants to do: reverse engineer the app code, attach a debugger, tamper with the app, etc. - the first step is usually to use an emulator. Only 4 out of 12 of the tested apps had such detection mechanisms in place.

75% of the apps are not capable of noticing whether it is being used in a hostile environment,

in which the basic security architectures of Android have been broken (as, for example, a rooted phone). A rooted device is much more at risk of being compromised, and therefore it is important to know about it. Detecting whether or not the device is running in a safe environment is essential for further security measures.

MORE THAN 80% OF THE APPS COULD BE REPACKAGED, INJECTED WITH MALWARE AND REDISTRIBUTED

When an app provider doesn't implement security to protect their app against repackaging, the app is highly compromised, and an attacker can easily modify it.

With the lack of proper security, bad actors can download a copy of the eGov app from the official distribution platforms (Google Play or App Store), modify or add malware to it without the app noticing such changes or foreign elements.

They can then redistribute this fake modified version of the app on official distribution platforms or other websites, where the app will be downloaded by users believing they are getting the original eGov app. One direct consequence could be that attackers scrape users' log-in credentials to access accounts and personal information and steal sensitive data.

The result of our research shows that 10 of the 12 apps analysed did not have the security in place that would protect against repackaging.

~ 60% OF THE TESTED APPS HAD NO MALWARE PROTECTION IN PLACE

Malware often exploits vulnerabilities and misuses the operating system features to gain access and steal users' personal data and credentials. Our study found that none of the eGov apps tested had a sufficient level of malware protection in place, and 60% of them had no malware protection at all.

Android's Accessibility Services provide apps with access to device settings and other programs. It is a crucial part of helping users with disabilities in using their device but also a common gateway for malware. By exploiting the Accessibility Services, malware can read the screen and log user inputs to extract highly sensitive data, such as personally identifiable information, passwords, and other credentials from apps.

When eGov apps don't protect against malware, they automatically put their citizens' data at risk.

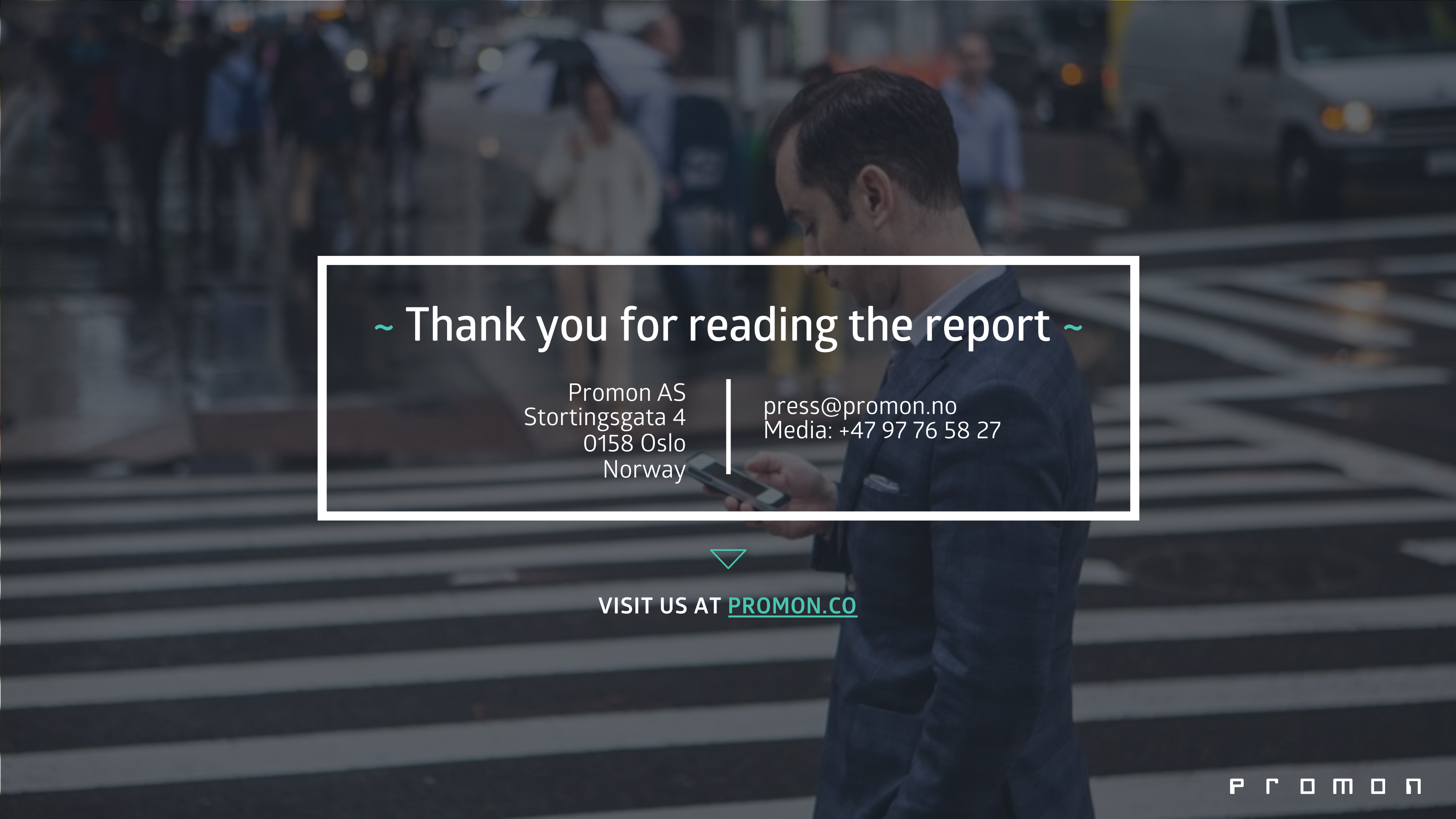
~ CONCLUSIONS AND SOME RECOMMENDATIONS

Arguably, all apps that hold sensitive information of their users have a responsibility to ensure that this information is kept safe. Our research on the selected apps in the eGov space in the APAC region shows that most of them have not implemented security mechanisms as recommended, for example, in the standards of OWASP. Consequently, the likelihood of data leaking or being manipulated is much higher when compared to an app that adheres to such standards.

Lacking resilience against commonly used attack tools and methods means that it requires very little technical skill or effort on the part of a bad actor for data to leak. Naturally, this increases the likelihood of such attacks and, because much of the data handled is sensitive, it will potentially have a significant impact on people if stolen.

To lower the risk of these vulnerabilities being identified and ultimately exploited, governments must adopt a comprehensive approach to app security - including App Shielding, strong encryption/data protection, and ensure their developers receive adequate secure programming training and implement security in the software development life cycle when writing the app code.

Most app users might expect that their data is being kept safe, and we believe that this trust and expectation should be taken seriously - especially by governmental apps.



~ Thank you for reading the report ~

Promon AS
Stortingsgata 4
0158 Oslo
Norway

press@promon.no
Media: +47 97 76 58 27

VISIT US AT PROMON.CO