TRUSTED FUTURE

DMA: TO PRESERVE ROBUST MOBILE SECURITY, POLICYMAKERS SHOULD HEED THE IMPORTANT SECURITY LESSONS FROM THE ACTIVEX ERA OF THE INTERNET

We've previously explained the many challenges European regulators face in implementing the Digital Markets Act (DMA), especially as it relates to the security consequences of downloading mobile apps from untrusted sources. In this piece, we go deeper to examine lessons from a similar well-meaning approach and architecture from the 1990's — a technology framework called ActiveX that enabled software applications to be downloaded from third-party sources to bridge the gap that separated web pages from Microsoft's operating system.

While powerful, ActiveX technologies would be assigned the highest security risk-level by the National Institute of Standards, became responsible for a broad range of major computer exploits, were called "unsecure by their very design," and eventually became so challenging that government security agencies around the globe found it necessary to warn users to stop using the technology altogether.

We explain why this was, and why we can't afford to rewind the clock or force the reintroduction of similar kinds of security issues that government cybersecurity experts, industry, and policymakers worked for more than a decade to eradicate. Instead, policymakers should heed the lessons of the past by driving robust privacy, safety, and security into the technology of tomorrow to make sure consumers, business, and governments can reap their full potential.

WHAT IS ACTIVEX?

History is filled with good intentions gone awry. First introduced in 1996, Microsoft's ActiveX was a technology that sought to expand user opportunity by connecting web pages to the power of the operating system by allowing third party programs to reach deeply into the brains and hardware of a user's computer. In its day, popular Internet Explorer ActiveX plug-ins were at the heart of the Internet's web-app ecosystem and enabled a broad range of games, video and audio players, and tools like Adobe Flash, Shockwave, Acrobat, and RealPlayer to run on the computer.

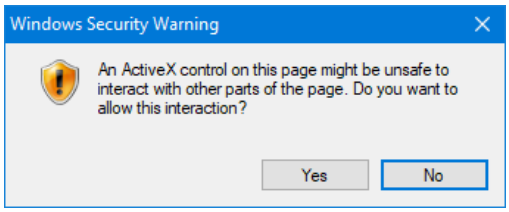
When visiting a website, Internet Explorer would automatically prompt the user to sideload and run any ActiveX application that the web page specified it wanted. While powerful, this model was unable to mitigate for a variety of security flaws, a long list of dangerous security exploits, and many of the era's most infamous computer viruses, spyware, zero-days, and adware programs which came disguised as ActiveX controls.

Like the sideloaded mobile app architecture envisioned by the DMA, ActiveX web apps were downloaded from unvetted third party sites, installed without a comprehensive security review, and given expansive access to the operating system and associated hardware.

THREE FOUNDATIONAL ACTIVEX SECURITY FLAWS

There were at least three fundamental architectural mistakes at the heart of the ActiveX model:

- First, ActiveX security relied entirely on human judgment. Unlike today's mobile apps downloaded from official app stores which are reviewed for security, each individual user had to decide which web-apps to install, and which ones were safe. When downloading the web app for the first time, a pop-up warned the user there could be security risks (the same kind of warning regulators are proposing users click through to sideload/download apps on mobile devices under the DMA). Users just needed to click yes to the dialog below:



  It puts the security burden on the user to vouch for the security of the app, who may not be aware of the security implications of the decision, without any trusted process the user could rely on to review or vet the app in advance. As expected, consumers did not have the security skills or background necessary to evaluate whether it was safe, and users commonly just clicked through the pop-up security warning – regardless of security impact–to enable the app functionality.

- The second primary mistake is, unlike mobile phones today, ActiveX gave the web apps almost unlimited access to the entirety of the underlying operating system, the computer's hardware, data, and even network resources, depending on the user's level of access. The executable binaries were not sandboxed in any way, so if you agreed to the pop-up, an ActiveX app didn't have restrictions for a set of safe actions but would instead be able to essentially do anything it wanted with all the files and programs on your computer. It's easy to see how ActiveX became a popular way to deliver malware and spyware. The DMA could require similar access under its provisions to grant unvetted apps access to the operating system, hardware and user data.

- Lastly, because these web-apps were downloaded outside of any centralized control system, unlike official app stores today, there was no way to update these sideloaded web apps. It meant that users often ended up running old, outdated, and exploitable code on their machines – creating new vectors for intrepid hackers. Once these ActiveX web-apps were installed on devices, keeping them up to date or getting rid of exploitable code altogether was often about as easy for the users as trying to put toothpaste back into the tube.

AS ONE CAN IMAGINE THIS APPROACH DIDN'T GO WELL – THE RESULTS WERE PREDICTABLE – AND MICROSOFT FOUND ITSELF ON A LONG JOURNEY ATTEMPTING TO ADDRESS KEY CHALLENGES

Not long after ActiveX was invented the Chaos Computer Club, a band of hackers from Germany, elucidated the consequences of embedding these vulnerabilities – by using ActiveX's access to the operating system to access Quicken software and steal money out of people's bank accounts. Soon, ActiveX exploits (and tricking users into downloading them) enabled bad actors to monitor personal browsing habits, install malware, generate adware pop-ups, log keystrokes and passwords, and do other malicious things. Hackers soon even figured out how to create ActiveX based zero day exploits to remotely execute almost any code, gain complete control of a computer, and even use ActiveX to remotely crash the computer.

As vulnerabilities came to light, Microsoft worked diligently to try to fix problems by adding features like "killbits" (2007) to prevent vulnerable ActiveX controls from running, ActiveX filtering (2011) to prevent sites from installing or using ActiveX, Protected Mode (2012) an attempt to isolate untrusted code, and along the way it also began warning users that ActiveX apps could be used to spy on your computer, damage data, give you content that you don't want like adware, install malicious software, or let someone else remotely control your computer.

Other major web browsers like Safari and Chrome took a fundamentally different security architecture approach to plug-ins by gate-keeping access to the platform and preventing untrusted apps from gaining unnecessary access to the operating system, hardware, or data.

Ultimately, faced with escalating and expansive ActiveX risks and exploits, governments around the globe eventually found the need to act by warning not to use ActiveX, nor the Internet Explorer web browser that was actively being exploited because of ActiveX. For example, in January of 2010, Germany's Federal Office for Information Security (BSI) began warning users not to use Internet Explorer, and to turn off ActiveX technology, because it allowed hackers to remotely plant and run malicious code on Windows PCs. Then in September of 2012, after discovering a widely used zero-day exploit and an "attempt to hijack one of the highly regarded German government servers through Internet Explorer," Germany's BSI told users to stop using Internet Explorer entirely, and Microsoft told users to block ActiveX controls. Days later France's government and Australia's government began warning users against using all versions of Internet Explorer. And as a result of the discovery of malware developed by Chinese hackers to exploit these vulnerabilities, the UK government recommended against Internet Explorer too. In 2014, because of significant vulnerabilities the U.S. government's Computer Emergency Readiness Team (US-CERT) also told users to stop using Microsoft's Internet Explorer browser because vulnerabilities could result in "the complete compromise" of an affected system.

It's worth highlighting that in 2014 when ENISA weighed in against the use of Internet Explorer, they argued that "there is no 100% security," and instead recommended that companies adopt an approach of "security-by-design." For the past 20 years, like ENISA, governments have been urging technology developers to adopt a "security-by-design" approach, and as newer products, like smartphones emerged, they began to adopt this approach too. Mobile security incorporates techniques like sandboxing, gatekeeping, vetting, and firewalling strategies to prevent untrusted code from third parties from being downloaded or executed.

Microsoft was never able to fully address the issues until 2015, when Microsoft replaced Internet Explorer with a fundamentally new architecture that was secure-by-design called the Edge browser – which specifically does not support ActiveX controls (although legacy users who need backwards compatibility can still temporarily run the legacy Internet Explorer mode.)

ARE WE FORGETTING THE GOOD AND HARD LESSONS LEARNED?

Some are now suggesting in the context of Europe's Digital Markets Act that instead of building upon the "security by design" approach that ENISA and leading experts from around the globe now recommend, that companies should instead be forced to go backwards on security. They suggest mobile operators should be forced to tear down the walls that have been built to prevent third party apps from gaining broad access to the mobile operating system, its data, and its associated hardware. Rather than embracing a model that allows for required and easy patching and protection, they would once again put the security burden on individual users by allowing the downloading and execution of unvetted code from untrusted third-party sources without a sandboxed system that comprehensively screens and can update buggy or malicious code.

We don't have to guess whether the same kind of issues that plagued the original ActiveX web-apps could be a problem in the mobile environment, we just need to look at the way the ability to sideload in the Android environment (apps downloaded from outside its official Play Store) that the DMA seemingly envisions expanding to the rest of the mobile ecosystem.

- For sideloaded Android apps downloaded outside of its official app store, **the security burden is once again transferred to the user and rooted in the user's judgement for third-party sideloaded apps**, just like ActiveX did, as shown in figure 1.

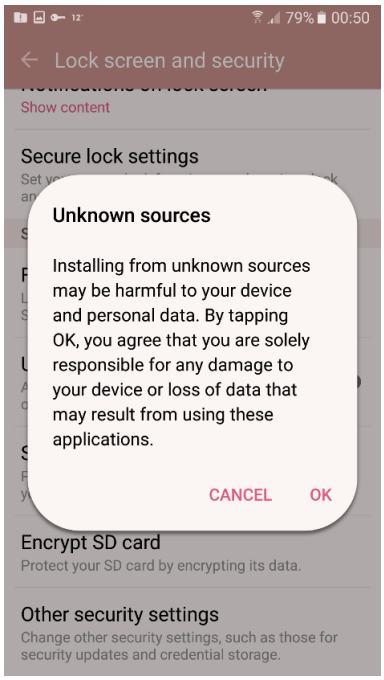
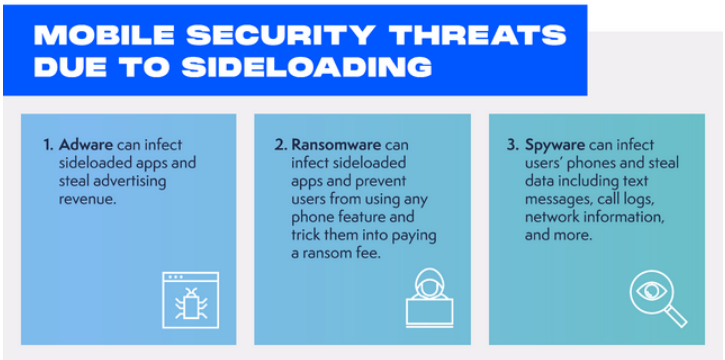


Figure 1: Android sideloading warning that similarly places entire security burden on the user.

- **The number of mobile security threats have sky-rocketed on platforms that support sideloading – unleashing new forms of spyware, adware, and ransomware.** In the same way that third-party ActiveX content created a broad range of new security threats, the same is true from third-party unvetted mobile apps downloaded from outside official app stores. The NOKIA Threat Intelligence Report, for example, found that devices that run on Android had 15 times more infections from malicious software than iPhone, with a key reason being that Android apps "can be downloaded from just about anywhere," while everyday iPhone users can only download apps from one source: the App Store. ENISA reported 230,000 new malware infections per day – translating to about 84 million per year – in 2019 and early 2020. These attacks spread malicious mobile malware, with some of today's most common types including adware, ransomware, spyware, banking, and other credential-stealing trojans masquerading as legitimate apps.



- **Adware.** For example, CopyCat malware spreads through popular apps repackaged to include malicious code and distributed through third-party stores and phishing scams. It infects devices in order to generate and steal advertising revenue. In just two months in 2016, CopyCat malware infected more than 14 million Android devices around the world.

- **Ransomware.** MalLocker.B is a family of Android malware distributed via sideloading. Once infected, it displays a ransom note over every other app window, ensuring that the target cannot use any other features of the phone. The hackers design the ransom note so that it looks like a legitimate notice from a local police department, informing the victims that they committed a crime and must pay a fine – providing instructions how to make payment.

- **Spyware.** FakeSpy is mobile malware that spies on users and steals their data and gets onto mobile devices that aren't protected against sideloading through a fake package delivery message (SMS Phishing.) It primarily affects Android users in France, Switzerland, Germany, the UK, the US, and Japan. A target receives a text message claiming that the postal service attempted to deliver a package, and that the user should track or sign for it. The message contains a link to a website that prompts users to sideload the fake delivery tracking app – at which point it requests permissions that allow it to obtain text messages, contact lists, call logs, network information, recently run tasks, and information about other apps.

- **Once again, security agencies from around the globe are warning against downloading apps from third-party sources.**

    - ENISA has consistently recommended that people *"Use the official application marketplace only... to minimize the risk of installing a malicious application.* ***Users should not sideload applications*** *if they do not originate from a legitimate and authentic source."*

    - This is nothing new. As early as 2010, ENISA began touting the benefits of ***"the 'walled garden' approach many smartphone vendors take to third-party software; by default, users can only add applications from a centrally controlled distribution channel." ...*** *"This means that app-store owners have the opportunity to perform a security review of apps before admitting them to the app-store and to remove apps from circulation which are subsequently shown to have security flaws.* ***Compared to other software distribution models and depending on the review process implemented, the walled-garden approach makes it more difficult for cyber attackers to spread malware" ...*** *"importantly, even if an attacker manages to get malware onto the marketplace, moderators can intervene at a later time, by removing the application from the market and thus limiting the number of affected users; controlled distribution limits the vectors or channels which can install malware on the device."*

    - Similarly, the European Union Agency for Law Enforcement Cooperation (Europol): "Just a game? **Only install apps from official app stores** . . . Be cautious of links you receive in email and text messages that might trick you into installing apps from third party or unknown sources."

    - Australia's eSafety Commissioner: "Apps for your iPhone or iPad should only ever be obtained from the App Store, while apps for Android devices should only be obtained from Google Play. Apps obtained from anywhere else may well be dangerous, and could try to misuse your information or put a virus on your phone."

POLICYMAKERS AND INNOVATORS SHOULD BE FOLLOWING THESE SECURITY AGENCIES' EXPERT ADVICE

At this critical moment in technological advancement, it's even more important today that we remember these lessons from the past. Regulatory changes aimed at forcing support for sideloading through direct downloads and third-party app stores would cripple the very privacy and security protections that are at the core of the most advanced mobile device protection systems and expose users to serious security risks. It would also enable and repeat the same kind of broad security challenges we saw from the original ActiveX era.

But this time, the impacts could be broader and solutions even more challenging to find. At a time when we are seeing an explosion in new AI tools, and a commensurate expansion in efforts by bad actors leveraging new tools in bad ways, this is not the time to turn back the clock on existing protections. New AI tools with impeccable grammar and capabilities are especially good at mimicking legitimate looking text, images, and apps. New tools like WormGPT that specialize in creating more effective phishing, and the targeted mobile smishing campaigns allow bad actors to launch sophisticated cyberattacks that trick users into giving up their personal information, inadvertently turning over their financial credentials, or downloading malicious apps.

It's of particular concern in the security world, since 90% of cyber-attacks start with and involve social engineering. Its why some are now warning that we could be hurtling toward a new "glitchy, spammy, scammy, AI-powered internet." As mobile devices have eclipsed laptops as the primary way we access the internet, they have also become a more valuable target. Because mobile devices aren't just relied upon by consumers to keep their information safe, but are also increasingly relied upon by enterprise networks, critical infrastructure environments, and by national security personnel, this is exactly the wrong time to be tearing down the guardrails, and reversing course on the "security by design" architecture. It's an architecture that relies upon multiple layers of security including regular software updates, strong authentication technology, restrictions on broad access to hardware and operating system functionality, and augments automatic systems with human review to weed out millions of untrustworthy applications. These existing mechanisms help protect against billions in losses from fraudulent and fake apps that could put our digital world at risk, and if not thwarted, could return us the kind of risky, scammy, unsafe original ActiveX type world. We should learn from the past, move forward, and should not force developers to make the same mistakes again.

A MORE TRUSTWORTHY PATHWAY FORWARD

Now that the DMA has passed, policymakers need to wade through the many ambiguities (and potential contradictory provisions) in how the DMA will be implemented to ensure it helps consumers and avoid the obvious and foreseeable problems of the past. Policymakers need to ensure they have sufficient technical expertise on hand necessary to engage in a meaningful regulatory dialogue that can swiftly identify and resolve the many important issues where they need to square the circle. They also would be wise to perform a comprehensive security threat assessment and technical review to align the dynamically changing threat environment, with the privacy and security protections consumers have come to expect from their leaders.

For other countries looking at replicating DMA-like proposals, like those being considered in Australia, Brazil, Canada, India, Japan, Turkey, and the U.K., they should be especially cautious, and pause their efforts until they can assure that by opening up the existing trusted app-ecosystem, they aren't actually opening up a new pandoras box full of broadly foreseeable problems. They should only proceed once these known issues have been resolved. We can't afford to rewind the clock and reintroduce the very same kind of problems that government cybersecurity experts and policymakers worked for more than a decade to eradicate. Instead, we need to drive robust privacy, safety, and security into our technology of tomorrow to make sure consumers, business, and governments can reap their full potential.