TESLA PATCHES FASTER THAN CHRYSLER … AND THAN ANDROID [UPDATED]

Wired's hack-of-the-day story reports that researchers hacked a Tesla (unlike the Chrysler hack, it required access to the vehicle once, though the Tesla also has a browser vulnerability that might not require direct access).

> Two researchers have found that they could plug their laptop into a network cable behind a Model S' driver's-side dashboard, start the car with a software command, and drive it. They could also plant a remote-access Trojan on the Model S' network while they had physical access, then later remotely cut its engine while someone else was driving.

The story notes how much more proactive Tesla was in patching this problem than Chrysler was.

> The researchers found six vulnerabilities in the Tesla car and worked with the company for several weeks to develop fixes for some of them. Tesla distributed a patch to every Model S on the road on Wednesday. Unlike Fiat Chrysler, which recently had to issue a recall for 1.4 million cars and mail updates to users on a USB stick to fix vulnerabilities found in its cars, Tesla has the ability to quickly and remotely deliver software updates to its vehicles. Car owners only have to click "yes" when they see a prompt asking if they want to install the upgrade.

In my understanding, Tesla was able to do this *both* because it responded right away to

implement the fix, *and* because it had the technical ability to distribute the update in such a way that was usable for end users. Chrysler deserves criticism for the former (though at least according to Chrysler, it *did* start to work on a fix right away, it just didn't implement it), but the latter is a problem that will take some effort to fix.

Which is one reason I think a better comparison with Tesla's quick fix is Google's delayed fix for the Stagefright vulnerability. As the researcher who found it explained, Google address the vulnerability *internally* immediately, just like Tesla did.

> Google has moved quickly to reassure Android users following the announcement of a number of serious vulnerabilities.
>
> The Google Stagefright Media Playback Engine Multiple Remote Code Execution Vulnerabilitiesallow an attacker to send a media file over a MMS message targeting the device's media playback engine, Stagefright, which is responsible for processing several popular media formats.
>
> Attackers can steal data from infected phones, as well as hijacking the microphone and camera.
>
> Android is currently the most popular mobile operating system in the world — meaning that hundreds of millions of people with a smartphone running Android 2.2 or newer could be at risk.
>
> Joshua Drake, mobile security expert with Zimperium, reports
>
> > A fully weaponized successful attack could even delete the message before you see it. You will only see the notification…Unlike spear-phishing, where the victim needs to open a PDF file or a link

> > sent by the attacker, this vulnerability can be triggered while you sleep. Before you wake up, the attacker will remove any signs of the device being compromised and you will continue your day as usual – with a trojaned phone.
>
> Zimperium say that “Google acted promptly and applied the patches to internal code branches within 48 hours, but unfortunately that’s only the beginning of what will be a very lengthy process of update deployment.”

But with Android the updates need to go through manufacturers, which creates a delay – especially given fairly crummy updating regimes by a number of top manufacturers.

The experience with this particular vulnerability may finally be pushing Android-based manufacturers to fix their update process.

> It’s been 10 days since Zimperium’s Joshua Drake revealed a new Android vulnerabilitycalled Stagefright – and Android is just starting to recover. The bug allows an attacker to remotely execute code through a phony multimedia text message, in many cases without the user even seeing the message itself. Google has had months to write a patch and already had one ready when the bug was announced, but as expected, getting the patch through manufacturers and carriers was complicated and difficult.
>
> But then, something unexpected happened: the much-maligned Android update system started to work. Samsung, HTC, LG, Sony and Android One have already announced pending patches for the bug, along with a device-specific patch for the Alcatel Idol 3. In Samsung’s case, the shift has

> kicked off an aggressive new security policy that will deploy patches month by month, an example that's expected to inspire other manufacturers to follow suit. Google has announced a similar program for its own Nexus phones. Stagefright seems to have scared manufacturers and carriers into action, and as it turns out, this fragmented ecosystem still has lots of ways to protect itself.

I make this comparison for two reasons. One, if Google — the customers of which have the hypothetical ability to send out remote patches, even if they've long neglected that ability — still doesn't have this fixed, it's unsurprising that Chrysler doesn't yet.

But some of the additional challenges that Chrysler has that Tesla has fewer of stem from the fragmented industry. Chrysler's own timeline of its vulnerability describes a "third party" discovering the vulnerability (not the hackers), and a "supplier" fixing it.

> In January 2014, through a penetration test conducted by a third party, FCA US LLC ("FCA US") identified a potential security vulnerability pertaining to certain vehicles equipped with RA3 or RA4 radios.
>
> A communications port was unintentionally left in an open condition allowing it to listen to and accept commands from unauthenticated sources. Additionally, the radio firewall rules were widely open by default which allowed external devices to communicate with the radio. To date, no instances related to this vulnerability have been reported or observed, except in a research setting.
>
> The supplier began to work on security improvements immediately after the

> penetration testing results were known in January 2014.

But it's completely unclear whether that "third party" is the "supplier" in question. Which means it's unclear whether this was found in the supplier's normal testing process or in something else.

One reason cars are particularly difficult to test are because so many different suppliers provide parts which don't get tested (or even adequately specced) in an integrated fashion.

Then, if you need to fix something you can't send out over a satellite or Internet network, you're dealing with the – in many cases – archaic relationships car makers have with dealers, not to mention the limitations of dealer staff and equipment to make the fix.

I don't mean to excuse the automotive industry – they're going to have to fix these problems (and the same problems lie behind fixing some of the defects tied to code that doesn't stem from hacks, too, such as Toyota's sudden acceleration problem).

It's worth noting, however, how simplified supply and delivery chains make fixing a problem a lot easier for Tesla than it is for a number of other entities, both in and outside of the tech industry.

UPDATE – 4:30 PM EDT –

Hey, it's Rayne here, adding my countervailing two cents (bitcoins?) to the topic after Marcy and I exchanged a few emails about this topic. I have a slightly different take on the situation since I've done competitive intelligence work in software, including open source models like Android.

Comparing Fiat Chrysler's and Google's Android risks, the size and scale of the exposures are a hell of a lot different. There are far more Android devices exposed than Chrysler car models at risk – +1 billion Android devices shipped

annually around the globe as of 4Q2014.

Hell, daily activations of Android devices in 2013 were 1.2 million devices per day – roughly the same number as all the exposed Chrysler vehicles on the road, subject to recall.

Google should have a much greater sense of urgency here due to the size of the problem.

Yet chances of a malware attack on an Android device actually causing immediate mortal threat to one or more persons is very low, compared to severity of Chrysler hack. Could a hacker tinker with household appliances attached via Android? It's possible – but any outcome now is very different from a hacker taking over and shutting down a vehicle operating at high speed in heavy traffic, versus shutting off a Phillips remote-controlled Hue lamp or a Google Nest thermostat, operating in the Internet of Things. The disparity in annoyance versus potential lethality may explain why Google hasn't acted as fast as Tesla – but it doesn't explain at all why Chrysler didn't handle announcing their vulnerability differently. Why did they wait nearly a year to discuss it in public?

Another substantial barrier for Google is the number of other moving parts when Android needs a security patch. If 81% of the entire smartphone market consisting of nearly 20,000 different devices, and +1 million corresponding Android-driven apps built for the same devices must be assessed and patched at the same time, complexity to secure the operating system is significantly greater than Chrysler's security patch. This is where Microsoft Window's closed proprietary model has a leg up on Google Android's open source model – where equipment was built to a monolithic standard and all software was centralized-top-down. Patches are more easily automated for release if all the equipment was built to accommodate a single operating system.

But Android's mobile service component, Google Mobile Services (GMS), has been customized by

device manufacturers to accommodate their equipment – several Chinese manufacturers have done this for phones used in their market. Korean device manufacturer Samsung's apps don't replace GMS, but operate alongside it while suppressing GMS' appearance to users. All Android devices rely on the underlying Android Open Source Platform (AOSP) codebase, released under a combination of Apache and GPL open source licenses. Google prevented dramatic forks in Android's code by releasing GMS as proprietary code, completely reliant on AOSP, discouraging licensing of AOSP to any manufacturer which did not also license GMS. (What manufacturers are willing to expend the resources needed to create entirely new mobile services based on AOSP? The cost is prohibitive except in markets the size of China.) Though most of the Android market is still AOSP+GMS code, how much of it will cooperate with a security patch push by Google? How much of the remaining AOSP+non-GMS devices can be secured, given their deviation from Google's Android?

All of this complicates Google's effort to secure Android. The next key difference between Android and Windows, further complicating security patching, is Android's 6-9 month refresh cycle, which is much faster and far more frequent than Windows ever was. The last refresh was Android Lollipop 5.1.x in late 2014; the next version, Android M, is tentatively scheduled for release 3Q2015. Should Google invest all its security efforts into Android M, and push all users to upgrade, or secure Lollipop now, or offer patches across the entire installed base from Android Froyo (circa 2010) up to a secured Android M? How does a company secure billions of devices running ten different versions of its operating system?

Perhaps Google is hurrying, as fast as it can given Android's installed base and the fur ball that is semi-open sourced licensing.

UPDATE – 5:00 PM EDT –

Oh. My. God.

Meet Certifi-gate, a newly disclosed Android vulnerability revealed by security firm Check Point at the Black Hat security conference in Las Vegas.

Perhaps Google is hurrying as fast as it can given the unrelenting firestorm raining down on its Android team.

_____

[*Caveat: Not only have I worked as a consultant in competitive intelligence for software companies, but I own GOOG and AAPL stock; my household has a 2014 Chrysler subject to security recall along with several Android-based devices. ~sigh~ / Rayne*]