Högskolan i Gävle

Sårbarheter i Apple iOS

Thomas Lundgren
thomaslundgren@live.com

2019-12-10

PM-uppgift på kursen Datakommunikation och IT-säkerhet, DVG324

Sammanfattning

Finurlig sammanfattning 😉.

Innehållsförteckning

1 Inledning

Apple, dess populära smarttelefon iPhone och surfplattan iPad, har länge ansetts vara de säkraste på marknaden, men år 2019 har varit det värsta hittills för Apples status som experter inom IT-säkerhet [1]. Då marknaden har översvämmats med ”zero-day-sårbarheter” i iPhone har värdet på iPhone-exploits sjunkit och är nu värderade lägre än motsvarande exploits för Android-enheter, enligt sårbarhetsmäklarna Zerodium och Crowdfense [2].

Flera av de sårbarheter som har uppmärksammats under 2019 har publicerats av Googles forskargrupp Project Zero. Den sjunde augusti publicerade Project Zero tio sårbarheter i iPhone [3] och den 29:e i samma månad publicerade forskargruppen ytterligare fem exploit-kedjor och 14 sårbarheter [4], varav sju i iPhones webbläsare Safari [5]. De fem exploit-kedjorna har utnyttjats för en enskild attack i över två år, i versionerna tio till tolv av iPhones operativsystem iOS och har givits särskild uppmärksamhet i media. Attacken påstås kunna vara den största attacken mot iPhone-användare hittills [6] och är fokuset i denna text.

Attacken gick ut på att implantera spionprogramvara (spyware) som hämtats från webbsidor genom iPhones webbläsare Safari. De fem exploit-kedjorna har sedan utnyttjas för att köra det skadliga programmet som kontaktar en server som ger instruktioner om vilka data som ska skickas från enheten till servern.

Förövarna har haft möjligheten att köra vilken skadlig kod de än önskat eftersom de har haft root-åtkomst i systemet. Att de valt att köra spionprogramvara som är ytterst svår för en vanlig användare att upptäcka lär vara en av anledningarna till att attacken har kunnat pågå i två års tid. Detta faktum, samt det att attacken misstänks ha riktats in på utvalda befolkningsgrupper [7], har lett säkerhetsexperter till att misstänka att attacken utförts av en regeringsmakt. Bland de misstänkta nationerna är bland annat Kina, Syrien och Israel då de alla har utsatta minoritetsgrupper som regeringsmakterna har ett stort intresse av att bevaka. Googles Project Zero har valt att inte avslöja vilka webbsidor som har infekterats med spionprogramvaran eller vilka befolkningsgrupper som har utsatts för attacken.

2 Förstudie om hot

2.1 Hot: Stöld av personliga data

Eftersom angriparna har haft root-åtkomst och har kunnat köra godtycklig kod på användarnas iPhone-enheter har hela systemet varit kompromitterat. Hot mot tillgänglighet, integritet och sekretess har funnits, men det verkar som att det endast är hotet mot sekretessen som har realiserats och det är därför det enda som diskuteras i denna text.

2.2 Tillgångar

Framför allt var det dessa data som angriparna var intresserade av [5]:

- Lösenord och autentiseringstokens lagrade i iCloud-nyckelringar,
- Meddelanden i diverse chattapplikationer (Skype, WhatsApp, Hangouts, iMessage, m.fl.),
- E-postmeddelanden i diverse e-postapplikationer (Gmail, Outlook, m.fl.),
- Samtalshistorik och SMS,
- Platsdata,
- Foton,
- m.m.

2.3 Potentiella skador

Stölden av dessa data kan ha många skadliga konsekvenser. På ett personligt plan kan det handla om att en persons rykte eller karriär sätts på spel. Det är inte heller otänkbart att en persons liv skulle kunna hotas under vissa omständigheter. I denna attack verkar det dock snarare som att informationen har använts för att kartlägga utvalda befolkningsgrupper.

3 Analys

Analysen nedan görs utifrån det välkända konceptet ”metod, möjlighet och motiv” (eng. method, opportunity and motive, MOM) [8].

3.1 Metod

Attacken var en så kallad ”vattenhålsattack” (eng. watering hole attack). Detta innebär att skadlig programvara placeras på en webbsida som många användare besöker. Användarnas enheter injiceras då med den skadliga programvaran. Den skadliga programvaran i det här fallet har endast varit riktad mot iPhone-användare. Attacken har pågått under över två år och angriparna har tvingats ändra sina metoder för att anpassa sig efter de nya iPhone-uppdateringarna som har gjorts. Exakt hur attacken har utförts i de olika iterationerna redovisas utförligt i Ian Beers blogginlägg, forskare vid Googles Project Zero [4].

Tre (den första, den tredje och den femte) av de fem exploit-kedjorna som utnyttjats för att utföra attacken redovisas nedan.

3.1.1 Exploit 1

Version(er) av iOS: 10.0.1–10.1.1
Sårbarhet: Sårbarheten fanns i en funktion i en GPU-drivrutin i iOS kärna [9]. Funktionen förutsatte att en av parametrarna hade en viss storlek, men ingen kontroll av detta gjordes i källkoden. Detta faktum utnyttjade angriparna för att utföra en ”heap overflow”-attack. Denna attack tillät angriparna att undkomma den sandlåda som kod som körs genom webbläsaren Safari befinner sig i och angriparna fick då root-åtkomst.

3.1.2 Exploit 3

Version(er) av iOS: 11.0–11.4
Sårbarhet: I en uppdatering introducerades en bugg i XPC, ett API för kommunikation mellan processer i iPhone-enheter [10]. Ett mindre än-tecken (`<`) byttes ut till ett ”inte lika med”-tecken (`!=`), vilket möjliggjorde att Safaris sandlåda kunde brytas ut ur.

3.1.3 Exploit 5

Version(er) av iOS: 11.4.1–12.1.2
Sårbarhet: Trots att angriparnas fjärde exploit fortfarande fungerade ändrade de strategi. Denna femte exploit var mer pålitlig och enklare att genomföra. Sårbarheten låg i ofärdig kod för en funktionalitet som Apple tänkt införa 2014, men som aldrig färdigställdes. Trots detta låg denna ofärdiga källkod kvar. Ian Beer, som publicerat denna exploit anmärker att ingen anställd hos Apple kan ha kört denna kod. De skulle då ha upptäckt att den får iPhone-enheten att krascha [11]:

> *To be clear, if there had been a test which called the syscall with the expected arguments, it would have caused a kernel panic. If any Apple developer had attempted to use this feature during those four years, their phone would have immediately crashed.*

Angriparna lyckades dock använda denna (förmodligen) otestade kod för att installera och köra sin spionprogramvara.

3.2 Möjlighet

Möjligheten att utföra denna attack kom ifrån sårbarheter i de API:er som Apple exponerar. I flera av de fem exploit-kedjorna som har utnyttjats verkar enkla fel som bör ha uppmärksammats och åtgärdats i utvecklingsfasen släppts till allmänheten.

3.3 Motiv

Motivet bakom attacken har som tidigare diskuterats förmodligen varit att kartlägga utsatta befolkningsgrupper. Eftersom syftet med kartläggningen är okänt kan vi endast spekulera kring vad som är motivet bakom den, men eftersom dåden mycket väl kan vara på uppdrag av en regeringsmakt kan de ha utförts i syfte att diskriminera, kontrollera eller fysiskt skada de utsatta.

4 Resultat

Svårt att diskutera tekniska lösningar, för dålig kunskap om iOS.

4.1 Motåtgärd 1: Testning och Quality Assurance (QA)

Den första motåtgärden som föreslås är en administrativ åtgärd, nämligen att kvaliteten och kraven på testning och QA ökar. Som tidigare redovisats har flera buggar som bör ha åtgärdats i testningsfasen möjliggjort denna attack.

4.2 Motåtgärd 2: Bug bounty program

Apple har ett ”bug bounty program” som ska locka white hat-hackers att rapportera sårbarheter mot betalning. Detta program har historiskt sätt verkat fungera bra då Apples enheter har varit kända som de säkraste på marknaden. På senare tid, och i synnerhet under 2019, har säkerheten minskat och buggarna ökat. Motåtgärden som föreslås är att öka kompensationen som utlovas för inrapporterade sårbarheter. Detta är något som Apple har genomfört [REF]

5 Referenser

[1] J. Cox, “This Has Been the Worst Year for iPhone Security Yet - VICE,” 2019. [Online]. Available: https://www.vice.com/en_ca/article/mbmgqp/this-is-worst-year-for-iphone-security-yet-2019. [Accessed: 12-Dec-2019].

[2] C. Miller, “iOS exploits ‘flood’ the security research market, experts say - 9to5Mac,” 2019. [Online]. Available: https://ww.9to5mac.com/2019/09/03/ios-exploit-market-report/. [Accessed: 12-Dec-2019].

[3] N. Silvanovich, “Project Zero: The Fully Remote Attack Surface of the iPhone,” 2019. [Online]. Available: https://googleprojectzero.blogspot.com/2019/08/the-fully-remote-attack-surface-of.html. [Accessed: 12-Dec-2019].

[4] I. Beer, “Project Zero: A very deep dive into iOS Exploit chains found in the wild,” 2019. [Online]. Available: https://googleprojectzero.blogspot.com/2019/08/a-very-deep-dive-into-ios-exploit.html. [Accessed: 12-Dec-2019].

[5] S. Golubev, “Websites have infected iPhones with spyware | Kaspersky official blog.” [Online]. Available: https://www.kaspersky.com/blog/malicious-websites-infect-iphones/28493/. [Accessed: 12-Dec-2019].

[6] J. Cox, “Google Says Malicious Websites Have Been Quietly Hacking iPhones for Years - VICE,” 2019. [Online]. Available: https://www.vice.com/en_us/article/bjwne5/malicious-websites-hacked-iphones-for-years. [Accessed: 12-Dec-2019].

[7] A. Greenberg and L. Hay Newman, “Mysterious iOS Attack Changes Everything We Know About iPhone Hacking | WIRED,” 2019. [Online]. Available: https://www.wired.com/story/ios-attack-watering-hole-project-zero/. [Accessed: 16-Dec-2019].

[8] C. P. Pfleeger and S. L. Pfleeger, *Analyzing Computer Security*. Westford: Paul Boger, 2012.

[9] I. Beer, “Project Zero: In-the-wild iOS Exploit Chain 1,” 2019. [Online]. Available: https://googleprojectzero.blogspot.com/2019/08/in-wild-ios-exploit-chain-1.html. [Accessed: 16-Dec-2019].

[10] I. Beer, “Project Zero: In-the-wild iOS Exploit Chain 3,” 2019. [Online]. Available: https://googleprojectzero.blogspot.com/2019/08/in-wild-ios-exploit-chain-3.html. [Accessed: 16-Dec-2019].

[11] I. Beer, “Project Zero: In-the-wild iOS Exploit Chain 5,” 2019. [Online]. Available: https://googleprojectzero.blogspot.com/2019/08/in-wild-ios-exploit-chain-5.html. [Accessed: 16-Dec-2019].