Bezpieczeństwo aplikacji mobilnych

Podręcznik hakera

Dominic Chell, Tyrone Erasmus
Shaun Colley, Ollie Whitehouse

Helion

Tytuł oryginału: The Mobile Application Hacker's Handbook

Tłumaczenie: Robert Górczyński

ISBN: 978-83-8322-548-7

Helion S.A.
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *https://helion.pl* (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
https://helion.pl/user/opinie/beapmv
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

Spis treści

Rozdział

1

(Nie)bezpieczeństwo aplikacji mobilnych

Nie ulega wątpliwości, że urządzenia mobilne zmieniły świat. W szczególności sposób pracy, a także współdziałania i komunikacji z innymi osobami nie będzie już więcej taki sam. Użytkownicy urządzeń mobilnych otrzymali wręcz nieskończone możliwości, które przez cały czas pozostają dostępne w zasięgu ich palców. Sprawdzenie salda konta bankowego, poczty elektronicznej, przeprowadzenie transakcji na giełdzie i znacznie więcej innych operacji jest po prostu na wyciągnięcie ręki. Opracowywanie aplikacji stało się na tyle popularnym zajęciem, że slogan Apple „Do tego celu jest aplikacja" nie odbiega od rzeczywistości.

W tym rozdziale dowiesz się, jak przebiegała ewolucja aplikacji mobilnych, oraz poznasz oferowane przez nie korzyści. Przedstawię również dane dotyczące podstawowych luk w zabezpieczeniach, które występują w aplikacjach mobilnych. Te dane pochodzą bezpośrednio z mojego doświadczenia i pokazują, że większość aplikacji mobilnych jest daleka od bezpiecznych. Następnie przejdę do skategoryzowania wspomnianych luk w zabezpieczeniach na podstawie opracowanej przez organizację OWASP (ang. *open web application security project*) listy dziesięciu największych zagrożeń dla aplikacji mobilnych. Zaprezentuję również ogólne omówienie wybranych narzędzi typu open source zalecanych przez OWASP przeznaczonych do sprawdzania bezpieczeństwa aplikacji mobilnych. Zobaczysz też, jak za pomocą tych narzędzi identyfikować pewne problemy wskazywane przez OWASP i gdzie ich szukać. Na końcu rozdziału przejdę do najnowszych trendów w zakresie zapewniania bezpieczeństwa aplikacjom mobilnym i tego, czego można w tym obszarze oczekiwać w przyszłości.

Ewolucja aplikacji mobilnych

Pierwsze aplikacje dla telefonów komórkowych zostały opracowane samodzielnie przez samych producentów urządzeń. Dla tych aplikacji istniała szczątkowa dokumentacja i można było znaleźć niewiele informacji dotyczących wewnętrznego sposobu działania tych programów. Przyczyną takiego stanu rzeczy były prawdopodobnie obawy producentów, że otwarcie platformy na aplikacje opracowane przez firmy zewnętrzne mogłoby doprowadzić do ujawnienia tajemnic jeszcze nie do końca dopracowanej technologii. Wczesne aplikacje były podobne do wielu dostarczanych przez producentów także w telefonach obecnej generacji, czyli na przykład książka adresowa, kalendarz i proste gry, takie jak popularna gra *Snake* stworzona przez Nokię.

Wraz z pojawieniem się smartfonów jako następców palmtopów (ang. *personal digital assistant*, PDA) tworzenie oprogramowania naprawdę nabrało rozpędu. Wzrost liczby dostępnych aplikacji dla urządzeń mobilnych mógł być też spowodowany bezpośrednio zwiększeniem mocy obliczeniowej oraz możliwości oferowanych przez smartfony, a także pochodzącym z rynku konsumenckiego wzrostem popytu na funkcjonalność. Gdy smartfony ewoluowały, aplikacje mobilne zaczęły wykorzystywać usprawnienia wprowadzane na platformach. Do wspomnianych usprawnień zaliczamy między innymi dodanie systemu nawigacji satelitarnej (ang. *global positioning system*, GPS), aparatu fotograficznego, zwiększenie czasu działania urządzenia na baterii, wprowadzenie lepszej jakości ekranów oraz procesorów. Te wszystkie ulepszenia spowodowały pojawianie się coraz bardziej rozbudowanych aplikacji, które dzisiaj znamy.

Opracowywanie aplikacji mobilnych przez firmy zewnętrzne urzeczywistniło się w 2008 roku, gdy firma Apple ogłosiła powstanie pierwszej usługi przeznaczonej do rozpowszechniania tego rodzaju aplikacji — App Store. Sklep App Store powstał rok po wypuszczeniu na rynek pierwszego smartfona Apple, czyli iPhone'a. Taki sam ruch wykonała firma Google ze swoim Android Market, który obecnie jest znany pod nazwą Google Play. Do dzisiaj pojawiło się wiele kolejnych sklepów oferujących aplikacje dla urządzeń mobilnych, między innymi Microsoft Store, Amazon Appstore i BlackBerry World.

Coraz większa konkurencja na rynku tworzenia aplikacji dla urządzeń mobilnych doprowadziła do pewnej fragmentacji rynków programistycznych. Większość aplikacji mobilnych jest przeznaczona dla konkretnej platformy, a twórcy oprogramowania są zmuszeni do pracy z różnymi systemami operacyjnymi, językami programowania i narzędziami, aby móc opracować programy dla wielu platform. Aplikacje na platformę iOS tradycyjnie były tworzone w języku Objective-C, aplikacje dla systemów Android i BlackBerry są tworzone w języku Java (w przypadku BlackBerry 10 również w Qt), natomiast aplikacje Windows Phone z użyciem .NET Framework. Tego rodzaju fragmentacja bardzo często prowadzi do konieczności istnienia w firmie tworzącej oprogramowanie dla urządzeń mobilnych wielu zespołów programistów oraz wielu baz kodu.

Jednak w ostatnim czasie na polu tworzenia aplikacji działających na różnych platformach mobilnych można dostrzec pewne zmiany, które wynikają z chęci zmniejszenia kosztów związanych z opracowywaniem programów. Niezależne od platformy frameworki oraz aplikacje oparte na standardzie HTML5 i działające w przeglądarkach WWW zyskały popularność dokładnie z wymienionych powodów. Można się spodziewać, że ten trend będzie coraz większy.

Najczęstsze kategorie aplikacji mobilnych

Aplikacje mobilne powstały dla praktycznie każdego możliwego do wyobrażenia sobie celu. Tylko w sklepach utrzymywanych przez Apple i Google znajduje się ponad 2 miliony aplikacji przeznaczonych do różnorodnych zastosowań, między innymi wymienionych poniżej:

- bankowość internetowa (Barclays),
- zakupy (Amazon),
- serwisy społecznościowe (Facebook),
- streaming (Sky Go),
- hazard (Betfair),
- komunikatory internetowe (WhatsApp),
- komunikatory głosowe (Skype),
- poczta elektroniczna (Gmail),
- współdzielenie plików (Dropbox),
- gry (*Angry Birds*).

Aplikacje mobilne bardzo często oferują funkcjonalność dostępną także za pomocą aplikacji internetowych. W wielu przypadkach używane jest to samo podstawowe API po stronie serwera, a widok przeznaczony dla smartfonów jest generowany na poziomie warstwy prezentacyjnej.

Poza aplikacjami dostępnymi w różnych sklepach istnieją jeszcze aplikacje, które zostały szeroko zaadaptowane w świecie biznesu i przeznaczone do zapewniania obsługi kluczowych zadań biznesowych. Wiele tego rodzaju aplikacji oferuje dostęp do ściśle strzeżonych danych korporacyjnych. Poniżej wymieniłem wybrane rodzaje programów, na które się natknąłem podczas wykonywania zadań zleconych przez klientów:

- Aplikacje magazynu dokumentów, pozwalające użytkownikom na uzyskanie dostępu na żądanie do ściśle strzeżonych dokumentów biznesowych.
- Aplikacje przeznaczone do prowadzenia małej księgowości, umożliwiające użytkownikom tworzenie, przechowywanie i przekazywanie do wewnętrznych systemów zestawień wydatków.
- Aplikacje HR, pozwalające użytkownikom na uzyskanie dostępu do listy płac, kart pracy, informacji o dniach wolnych oraz innych danych, które można uznać za poufne.
- Aplikacje usług wewnętrznych, takie jak zoptymalizowane pod kątem zapewniania dostępu do zasobów wewnętrznych, na przykład intranetu korporacyjnego.
- Aplikacje wewnętrznych komunikatorów, umożliwiające użytkownikom komunikację w czasie rzeczywistym niezależnie od ich położenia.

We wszystkich wymienionych powyżej przykładach aplikacje są uznawane za „wewnętrzne" i zwykle są opracowywane przez daną organizację lub specjalnie na jej zlecenie. Dlatego też wiele tych aplikacji wymaga użycia wirtualnej sieci prywatnej (ang. *virtual private network*, VPN) lub dostępu do sieci wewnętrznej, aby mogły prawidłowo funkcjonować wraz z wewnętrzną infrastrukturą. Zyskującym coraz większą popularność trendem w aplikacjach korporacyjnych jest

wprowadzenie technologii „geo-fence", w której to aplikacja używa wbudowanego w urządzenie GPS-u w celu sprawdzenia, czy użytkownik na pewno znajduje się w określonej lokalizacji, na przykład biurze firmy, a następnie na podstawie wyniku tego sprawdzenia udziela dostępu lub nakłada ograniczenia funkcjonalności.

Zalety aplikacji mobilnych

Nietrudno dostrzec, dlaczego aplikacje mobilne nabrały tak dużego znaczenia w stosunkowo krótkim okresie. Bodźce komercyjne i korzyści przynoszone przez aplikacje mobilne są oczywiste. Dają organizacjom możliwość dotarcia do użytkowników końcowych praktycznie w każdej chwili, a ze względu na ogromną popularność smartfonów trafiają do znacznie większej grupy docelowej. Jednak na sukces aplikacji mobilnych wpływ miało również wiele innych czynników, z których wybrane wymieniłem poniżej:

- Fundamenty dla aplikacji mobilnych powstały na bazie istniejących i popularnych protokołów. W szczególności protokół HTTP jest doskonale znany przez programistów i dlatego pozostaje dość powszechnie stosowany we wdrożeniach mobilnych.
- Techniczny rozwój smartfonów pozwolił aplikacjom mobilnym na zaoferowanie znacznie bardziej zaawansowanych funkcji oraz lepszych wrażeń, jakie odnoszą użytkownicy tych aplikacji. Usprawnienia wprowadzone w zakresie rozdzielczości ekranu oraz pojawienie się ekranów dotykowych stanowiły najważniejsze czynniki wpływające na poprawę interaktywności programu, szczególnie w grach. Wydłużenie czasu działania urządzenia na baterii oraz zwiększenie mocy obliczeniowej pozwalają nowoczesnym smartfonom na jednoczesne uruchomienie nie tylko jednej, ale wielu aplikacji, które na dodatek będą działały znacznie dłużej. To oznacza doskonałe udogodnienie dla użytkowników końcowych, ponieważ otrzymują pojedyncze urządzenie, które może służyć do wielu celów.
- Usprawnienia wprowadzone w technologiach sieci komórkowych doprowadziły do znacznego wzrostu szybkości komunikacji. W szczególności powszechny zasięg 3G i 4G pozwala użytkownikom na szerokopasmowy dostęp do internetu w smartfonach. Aplikacje mobilne w pełni wykorzystują zalety szybkiej komunikacji, oferując dostęp do naprawdę wielu usług internetowych.
- Prostota podstawowych technologii oraz języków programowania używanych do tworzenia aplikacji mobilnych pomogła w rewolucji mobilnej. Programy mogą być tworzone za pomocą popularnych i dopracowanych języków, takich jak Java, które są doskonale znane i mają ogromne bazy użytkowników.

Bezpieczeństwo aplikacji mobilnych

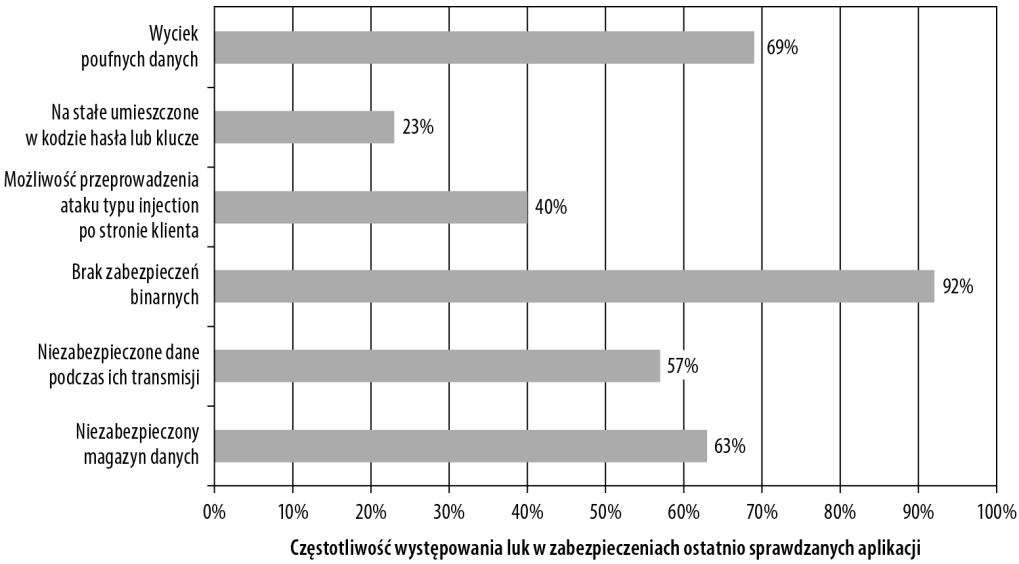
Aplikacje mobilne są narażone na istnienie wielu różnych luk w zabezpieczeniach, z których część jest dziedziczona po tradycyjnych atakach zarówno na aplikacje internetowe, jak i zwykłe programy komputerowe. Jednak kilka innych klas ataków jest charakterystycznych wyłącznie dla platform mobilnych i pojawia się jako efekt sposobu użycia aplikacji mobilnych i na skutek istnienia względnie unikatowych punktów wejścia i płaszczyzn ataku tworzonych przez te aplikacje.

Poniżej wymieniłem możliwe płaszczyzny ataku na aplikacje mobilne — programiści powinni mieć świadomość ich istnienia i potrafić się przed nimi bronić.

- Większość aplikacji mobilnych prowadzi pewnego rodzaju komunikację sieciową. Ze względu na sposób używania urządzeń mobilnych wspomniana komunikacja może odbywać się przez niezaufane lub niebezpieczne sieci, takie jak Wi-Fi udostępniane przez hotel lub lokal gastronomiczny, mobilne hot-spoty lub sieć komórkowa. Jeśli dane nie są odpowiednio zabezpieczone podczas ich przesyłania, aplikacja może być narażona na wiele różnych niebezpieczeństw, między innymi ujawnienie poufnych danych i ataki typu injection.
- Skoro użytkownik nosi ciągle przy sobie urządzenie mobilne, rodzi to naprawdę wiele możliwości jego zagubienia lub kradzieży. Programiści aplikacji mobilnych muszą zdawać sobie sprawę z niebezpieczeństwa, jakie wiąże się z próbami odzyskania danych z systemu plików urządzenia. Każda trwała treść pozostawiana przez aplikację w systemie plików — niezależnie od tego, czy w trwałym magazynie danych, czy w buforze tymczasowym — może potencjalnie doprowadzić do ujawnienia atakującemu poufnych danych.
- Dość unikatowym dla aplikacji mobilnych scenariuszem jest zagrożenie płynące ze strony samego urządzenia. Złośliwie działające malware szerzy się na platformie mobilnej, co przynajmniej po części wynika z istnienia nieoficjalnych kanałów dystrybucji oprogramowania. Programiści muszą zdawać sobie sprawę z niebezpieczeństwa, jakim może być atak ze strony innej aplikacji zainstalowanej w urządzeniu mobilnym.
- Aplikacje mobilne mogą pobierać dane wejściowe z wielu różnych możliwych źródeł, co powoduje utworzenie znacznej liczby potencjalnych punktów wejścia. Na przykład nietrudno natknąć się na aplikację akceptującą dane z jednego lub wielu wymienionych źródeł: NFC (ang. *near field communication*), Bluetooth, aparat fotograficzny, mikrofon, SMS (ang. *short message service*), USB (ang. *universal serial bus*), kody QR (ang. *quick response*) itd.

Do najpoważniejszych ataków na aplikacje mobilne zaliczamy te, które powodują ujawnienie poufnych danych lub też ułatwiają włamanie do urządzenia. Tego rodzaju luki w zabezpieczeniach są najczęściej związane z urządzeniem mobilnym i danymi użytkownika końcowego, a dużo rzadziej z wszystkimi użytkownikami danej usługi. Wprawdzie istniejące po stronie serwera luki w zabezpieczeniach stanowią największe zagrożenie dla wdrożeń aplikacji mobilnych jako całości, ponieważ mogą umożliwić niczym nieograniczony dostęp do systemów back endu, ale te problemy są doskonale udokumentowane i rozumiane przez programistów. Istniejące po stronie serwera luki w zabezpieczeniach wpływające na aplikacje mobilne nie będą omawiane w tej książce. Jeżeli chcesz dowiedzieć się więcej na temat tego rodzaju ataków, zachęcam Cię do zapoznania się z książką *The Web Application Hacker's Handbook* (*http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1118026470.html*).

Kwestie bezpieczeństwa aplikacji mobilnych nadal są błędnie postrzegane, choć z drugiej strony ten obszar nie jest jeszcze w pełni dojrzały. Tak naprawdę większość aplikacji mobilnych wciąż można uznać za niewystarczająco zabezpieczone. Po przetestowaniu w ostatnich latach setek aplikacji mobilnych w większości z nich znajdowałem jeden lub więcej poważnych błędów w zabezpieczeniach. Na rysunku 1.1 pokazałem procentowe dane dotyczące przetestowanych od 2012 roku aplikacji mobilnych, w których znalazłem pewne najczęściej spotykane kategorie luk w zabezpieczeniach po stronie klienta.



Rysunek 1.1. Częstotliwość występowania luk w zabezpieczeniach aplikacji ostatnio sprawdzanych przez autora

- **Niezabezpieczony magazyn danych (63%)**. Ta kategoria luk w zabezpieczeniach obejmuje różne defekty, które prowadzą do tego, że aplikacja przechowuje w urządzeniu mobilnym dane w postaci zwykłego tekstu. Z reguły jest stosowany zaciemniony format wykorzystujący na stałe umieszczony w kodzie klucz lub też inne rozwiązanie pozwalające osobie atakującej w dość łatwy sposób odczytać informacje przechowywane w magazynie danych.
- **Niezabezpieczone dane podczas ich transmisji (57%)**. Ta kategoria obejmuje każdy przypadek, gdy aplikacja nie używa szyfrowania warstwy transportu w celu ochrony danych podczas ich transmisji. To obejmuje także sytuacje, w których wykorzystano szyfrowanie warstwy transportu, choć sam mechanizm szyfrowania został zaimplementowany w niewystarczająco bezpieczny sposób.
- **Brak zabezpieczeń binarnych (92%)**. Ten defekt oznacza, że aplikacja nie stosuje żadnej formy mechanizmu obronnego utrudniającego przeprowadzenie procesu inżynierii odwrotnej, złośliwe manipulowanie programem lub jego debugowanie.
- **Możliwość ataku typu injection po stronie klienta (40%)**. Ta kategoria luk w zabezpieczeniach dotyczy scenariuszy, w których niezaufane dane wejściowe są przekazywane aplikacji, gdzie następnie będą przetwarzane w niebezpieczny sposób. Typowe źródła ataków typu injection obejmują na przykład inne aplikacje zainstalowane w urządzeniu oraz pochodzące z serwera dane wejściowe przekazywane do aplikacji.
- **Na stałe umieszczone w kodzie hasła lub klucze (23%)**. Ten problem powstaje, gdy programista na stałe umieszcza w kodzie aplikacji pewne informacje wrażliwe, takie jak hasło lub klucz szyfrowania.
- **Wyciek poufnych danych (69%)**. Ta kategoria obejmuje sytuacje, w których aplikacja przypadkowo ujawnia poufne dane w wyniku ataku typu *side-channel*, polegającego na utworzeniu dodatkowego kanału informacyjnego. W szczególności mam tutaj na myśli następujący bez wiedzy programisty wyciek danych na skutek użycia pewnego frameworka lub systemu operacyjnego.

Kluczowe czynniki problemu

Problemy związane z bezpieczeństwem aplikacji mobilnych pojawiają się z wielu różnych powodów. Z kolei luki w zabezpieczeniach zwykle występują, gdy aplikacja musi obsługiwać lub chronić poufne dane bądź też przetwarzać dane pochodzące z niezaufanych źródeł. Jednak połączenie kilku innych czynników może jeszcze bardziej spotęgować problem.

Niewystarczająca świadomość dotycząca zapewniania bezpieczeństwa

W przeciwieństwie do większości aplikacji internetowych, w których płaszczyzna ataku jest ograniczona do pochodzących od użytkownika danych wejściowych, programiści tworzący aplikacje mobilne muszą uwzględnić wiele różnych sytuacji powodujących problemy i odpowiednio chronić program przed nimi. W porównaniu do budowy innego rodzaju aplikacji, tworzenie aplikacji mobilnych jest dość unikatowe z tego względu, że nie można ufać systemowi operacyjnemu urządzenia, a nawet własnej aplikacji. Świadomość istnienia wielu płaszczyzn ataku i sposobów obrony przed nimi jest dość ograniczona i właściwie słabo rozpowszechniona w społeczności programistów tworzących tego rodzaju aplikacje. W kwestii zapewniania bezpieczeństwa mobilnego nadal mamy do czynienia z szeroko rozpowszechnioną dezorientacją i błędnym pojmowaniem podstawowych koncepcji. Jako przykład można tutaj przedstawić przekonanie wielu programistów o braku konieczności szyfrowania lub ochrony danych trwale przechowywanych w urządzeniu. Ci programiści tłumaczą to istnieniem szyfrowania standardowo zapewnianego przez wiele urządzeń. Jak się później przekonasz, to założenie jest niewłaściwe i może doprowadzić do ujawnienia poufnych danych użytkownika.

Nieustannie zmieniające się płaszczyzny ataku

Badanie poziomu bezpieczeństwa urządzeń mobilnych i aplikacji to nieustannie ewoluująca dziedzina, w której regularnie są odkrywane nowe zagrożenia i koncepcje. Szczególnie w przypadku urządzeń odkrywane są nowe luki w zabezpieczeniach, które mogą podważyć powszechnie przyjęte i stosowane w aplikacjach strategie ich ochrony. Jako przykład można tutaj podać odkrycie w urządzeniach Apple luki o nazwie „goto fail" (*https://support.apple.com/en-us/HT202934*), osłabiającej spójność tego, co wcześniej było uznawane za bezpieczny kanał komunikacji. W przypadku tej luki można było obejść nawet zalecane środki bezpieczeństwa, takie jak przypinanie certyfikatu. To doprowadziło do tego, że wielu programistów i ludzi profesjonalnie zajmujących się kwestiami zapewniania bezpieczeństwa zaczęło szukać drugiego schematu szyfrowania i implementować go w celu ochrony danych przepływających przez kanał SSL i TLS. Tego rodzaju luki w zabezpieczeniach potwierdzają, że nieustanne badania mogą wpływać na profil zagrożenia dla aplikacji lub go zmieniać, nawet w trakcie prowadzenia już prac nad nią. Zespół programistów rozpoczynający pracę nad projektem i posiadający dokładną wiedzę na temat aktualnych zagrożeń może być zmuszony do wprowadzenia zmian w kwestii zapewniania bezpieczeństwa, jeszcze zanim aplikacja zostanie ukończona i wydana.

Ograniczenia ekonomiczne i czasowe

Większość projektów programistycznych boryka się ze ściśle określonymi ograniczeniami zasobów i czasu; projekty aplikacji mobilnych nie są pod tym względem wyjątkiem. Kwestie ekonomiczne podczas tworzenia aplikacji bardzo często oznaczają, że zatrudnienie na stałe eksperta z dziedziny

bezpieczeństwa jest dla firmy po prostu niewykonalne. To dotyczy zwłaszcza mniejszych organizacji, w których testowanie pod kątem zapewniania bezpieczeństwa zwykle jest przeprowadzane na bardzo późnym etapie cyklu życiowego projektu. Nie ulega wątpliwości, że mała firma zwykle ma dużo mniejszy budżet, a tym samym charakteryzuje się mniejszą skłonnością do ponoszenia wydatków na usługi konsultingowe w zakresie zapewniania bezpieczeństwa. Wprawdzie krótki test penetracyjny prawdopodobnie pozwoli na wykrycie najprostszych luk w zabezpieczeniach, ale z reguły nie doprowadzi do wskazania znacznie subtelniejszych i skomplikowanych problemów, które można odkryć, mając więcej czasu i cierpliwości. Nawet jeśli podczas pracy nad projektem jest stale dostępny ekspert z dziedziny bezpieczeństwa, to ścisłe ograniczenia czasowe mogą uniemożliwić dokładną analizę każdego wydania aplikacji. Stosowanie zwinnych (ang. *agile*) metod programowania, w których mamy do czynienia z wieloma iteracjami w krótkim okresie, może jeszcze bardziej spotęgować ten problem.

Niestandardowy proces tworzenia aplikacji

Aplikacje mobilne są zwykle tworzone przez programistów danej organizacji lub zespoły zewnętrzne. W pewnych przypadkach stosowane jest połączenie obu wymienionych podejść. Ogólnie rzecz biorąc, jeśli organizacja regularnie opracowuje wiele aplikacji, doskonale przetestowane komponenty będą wielokrotnie używane w różnych projektach, co bardzo często prowadzi do promocji znacznie bardziej solidnego i bezpiecznego kodu. Jednak nawet jeśli aplikacje wykorzystują już istniejące komponenty z innych projektów, to i tak bardzo często w tych programach pojawiają się biblioteki lub frameworki, które nie zostały opracowane przez programistów budowanej aplikacji. W takim przypadku twórcy projektu mogą nie mieć pełnej i dokładnej wiedzy dotyczącej używanego kodu, co może z kolei doprowadzić do powstania problemów związanych z zapewnianiem bezpieczeństwa. Co więcej, w niektórych przypadkach biblioteki również mogą zawierać luki w zabezpieczeniach, jeśli nie zostaną dokładnie przetestowane. Przykładem może być tutaj luka `addJavascriptInterface`, wpływająca na komponent `Webview` w systemie Android. Po jej wykorzystaniu osoba atakująca może zdalnie włamać się do urządzenia. Dalsze badania nad tą luką pozwoliły określić, że występowała ona w bibliotekach używanych w celu zapewnienia integracji z reklamami i potencjalnie mogła wpływać na ogromną liczbę aplikacji (*https://labs.mwrinfosecurity.com/blog/webview-addjavascript-interface-remote-code-execution/*).

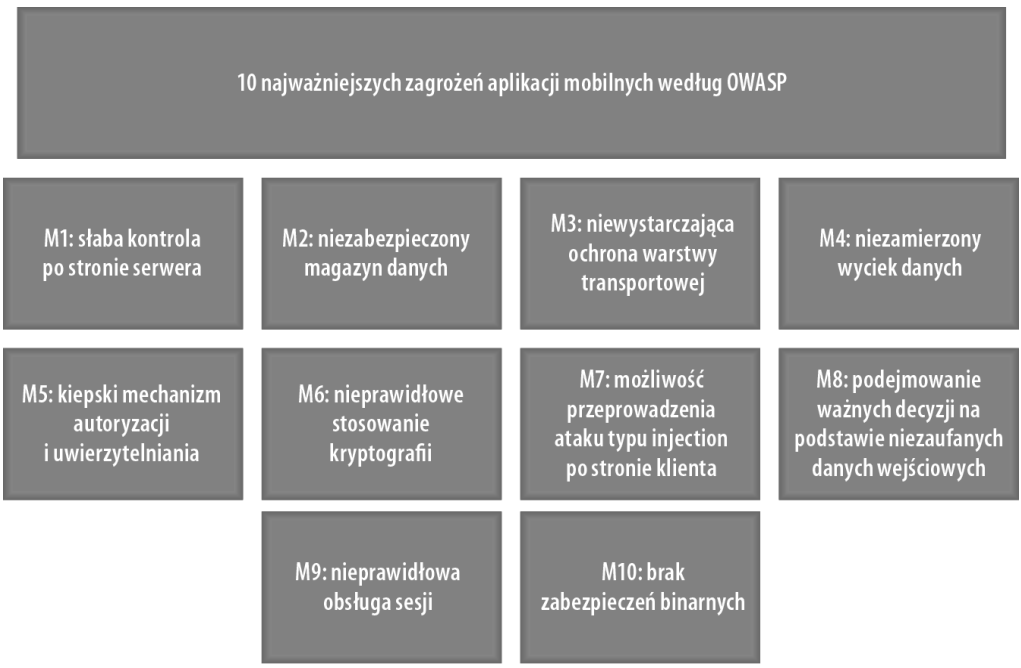
Projekt OWASP Mobile Security

Projekt OWASP Mobile Security (*https://www.owasp.org/index.php/OWASP_Mobile_Security_Project*) to inicjatywa utworzona przez OWASP, czyli grupę typu non profit, która jest doskonale znana ze swojej pracy nad kwestiami związanymi z zapewnianiem bezpieczeństwa aplikacji internetowych. Z uwagi na wiele podobieństw istniejących między aplikacjami mobilnymi i internetowymi naturalne jest to, że grupa OWASP stara się promować i zwiększać świadomość w zakresie problemów dotyczących bezpieczeństwa mobilnego.

Projekt oferuje bezpłatne, scentralizowane źródło klasyfikowania ryzyka w zakresie bezpieczeństwa mobilnego. Ponadto dokumentuje działania programistyczne, które mogą zmniejszyć wpływ lub prawdopodobieństwo wykorzystania danej luki w zabezpieczeniach. Projekt jest skoncentrowany na warstwie aplikacji, a nie bezpieczeństwie platformy mobilnej. Jednak pod uwagę jest brane również ryzyko nierozerwalnie związane z użyciem poszczególnych platform mobilnych.

10 najważniejszych zagrożeń aplikacji mobilnych według OWASP

Podobnie jak w przypadku dziesięciu największych zagrożeń OWASP, twórcy projektu Mobile Security zdefiniowali odpowiadających im dziesięć najważniejszych problemów związanych z bezpieczeństwem na platformach mobilnych. W tym zakresie projekt dość szeroko kategoryzuje pewne największe zagrożenia. Teraz pokrótce podsumuję pozycje wymienione na tej liście OWASP. Znacznie bardziej szczegółowe omówienie poszczególnych problemów i wskazówki dotyczące ich rozwiązań znajdziesz na stronie wiki projektu pod adresem *https://www.owasp.org/index.php/OWASP_Mobile_Security_Project#tab=Top_10_Mobile_Risks*. Graficznie dziesięć najważniejszych problemów według OWASP pokazałem na rysunku 1.2.



Rysunek 1.2. Wymienione przez grupę OWASP dziesięć najważniejszych problemów związanych z bezpieczeństwem na platformach mobilnych

Oto krótkie omówienie dziesięciu największych zagrożeń aplikacji mobilnych według projektu OWASP Mobile Security:

- **M1: słaba kontrola po stronie serwera**. Ta kategoria ryzyka została oceniona jako najpoważniejsze zagrożenie dla aplikacji mobilnych. Wpływ tej kategorii został określony jako niezwykle ciężki, poważny błąd w kontroli po stronie serwera, który może mieć istotne konsekwencje dla biznesu. Ta kategoria obejmuje wszelkie luki w zabezpieczeniach, które mogą występować po stronie serwera, między innymi w usługach sieciowych, konfiguracjach serwerów WWW oraz w tradycyjnych aplikacjach internetowych. Umieszczenie tej kategorii na liście najpoważniejszych zagrożeń dla platformy mobilnej można uznać za kontrowersyjne, ponieważ problem nie występuje w urządzeniu mobilnym. Istnieją oddzielne projekty, które wyraźnie zajmują się zagrożeniami aplikacji internetowych. Wprawdzie zdaję sobie

sprawę z wagi tego zagrożenia, ale ta kategoria nie jest dokładnie omówiona w książce, ponieważ została już doskonale udokumentowana w innych publikacjach, na przykład we wspomnianej wcześniej książce zatytułowanej *The Web Application Hacker's Handbook* (*http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1118026470.html*).

- **M2: niezabezpieczony magazyn danych**. Ta kategoria ryzyka wiąże się z sytuacją, gdy aplikacja przechowuje w urządzeniu mobilnym poufne dane w formacie zwykłego tekstu lub innej postaci łatwej do odczytania przez osobę atakującą. Wpływ tej kategorii został określony jako niezwykle ciężki, poważny błąd, który zwykle może mieć poważne konsekwencje dla biznesu, na przykład doprowadzi do kradzieży tożsamości, oszustwa lub utraty reputacji. Poza umożliwieniem fizycznego dostępu do urządzenia, niebezpieczeństwo wykorzystania luki w tej kategorii wiąże się również z uzyskaniem dostępu do systemu plików, co może nastąpić poprzez użycie oprogramowania typu malware lub złamanie zabezpieczeń urządzenia w inny sposób.
- **M3: niewystarczająca ochrona warstwy transportowej**. Ta kategoria ryzyka odnosi się do ochrony ruchu sieciowego i może pojawić się w każdej sytuacji, w której dane są przekazywane w postaci zwykłego tekstu. Może dotyczyć także scenariuszy, w których ruch jest szyfrowany, choć samo szyfrowanie zostało zaimplementowane w nieprawidłowy sposób, na przykład przez dopuszczenie użycia samodzielnie podpisanych certyfikatów, przeprowadzenie niewystarczającej weryfikacji certyfikatów lub użycie niebezpiecznych pakietów szyfrowania. Tego rodzaju problemy mogą zostać wykorzystane przez osobę atakującą znajdującą się w sieci lokalnej lub sieci operatora; nie jest wymagany fizyczny dostęp do urządzenia.
- **M4: niezamierzony wyciek danych**. Tego rodzaju problem pojawia się, gdy programista przypadkowo umieszcza poufne informacje lub dane w miejscach urządzenia mobilnego, które są łatwo dostępne z poziomu innych aplikacji. Bardzo często tego rodzaju zagrożenia pojawiają się jako efekt uboczny danej platformy, przede wszystkim na skutek braku wystarczającej wiedzy programistów odnośnie sposobów przechowywania danych przez system operacyjny. Często spotykałem się z przykładami niezamierzonego wycieku danych, między innymi z powodu stosowania buforowania, migawek i dzienników zdarzeń aplikacji.
- **M5: kiepski mechanizm autoryzacji i uwierzytelniania**. Ta kategoria ryzyka jest związana z błędami popełnianymi podczas uwierzytelniania i autoryzacji, które mogą występować w aplikacji mobilnej lub w implementacji działającej po stronie serwera. Lokalne uwierzytelnianie w aplikacji mobilnej jest dość powszechne, szczególnie w aplikacjach zapewniających dostęp do poufnych danych i wymagających działania w trybie offline. W przypadku braku zastosowania odpowiednich środków bezpieczeństwa istnieje ryzyko obejścia mechanizmu uwierzytelniania i tym samym uzyskania dostępu do aplikacji. Tego rodzaju niebezpieczeństwo wiąże się również z błędami autoryzacji, które mogą występować w aplikacji działającej po stronie serwera. Skutkiem jest wówczas umożliwienie uzyskania dostępu lub wykonanie pewnych funkcji ponad uprawnieniami przydzielonymi danemu użytkownikowi.

- **M6: nieprawidłowe stosowanie kryptografii**. Powszechnie przyjętą koncepcją jest to, że aplikacja przechowująca dane w urządzeniu mobilnym powinna je szyfrować w celu zapewnienia poufności tych danych. Stosowanym w takich przypadkach rozwiązaniem jest szyfrowanie, ale ryzyko pojawia się, gdy jego implementacja zawiera pewne słabe strony. W najgorszym przypadku osoba atakująca może uzyskać fragmenty danych w postaci zwykłego tekstu lub nawet oryginalne dane w niezaszyfrowanej postaci. Najczęściej niebezpieczeństwo pojawia się na skutek kiepskiego procesu zarządzania kluczem szyfrowania. Przykładem może być tutaj umieszczenie klucza prywatnego w aplikacji, umieszczenie na stałe w kodzie klucza statycznego lub użycie klucza, który może być w niezwykle łatwy sposób znaleziony w urządzeniu — jak użycie jako klucza po prostu identyfikatora urządzenia Android.
- **M7: możliwość przeprowadzenia ataku typu injection po stronie klienta**. Ataki typu injection mogą występować, gdy aplikacja mobilna akceptuje dane wejściowe pochodzące z niezaufanych źródeł. To mogą być źródła wewnętrzne dla urządzenia, na przykład inna aplikacja, lub też zewnętrzne, takie jak komponent działający po stronie serwera. Rozważ przykład aplikacji społecznościowej pozwalającej użytkownikom na publikowanie komunikatów zawierających różne informacje o tym, co się w danej chwili dzieje z daną osobą. Aplikacja mobilna pobiera z witryny internetowej opublikowane przez innych użytkowników komunikaty, które następnie wyświetla. Jeżeli osoba atakująca będzie w stanie utworzyć fałszywy komunikat przechowywany we wspomnianej witrynie internetowej i pobrany później przez innego użytkownika aplikacji mobilnej, wówczas po umieszczeniu tego komunikatu w komponencie widoku sieciowego bądź w bazie danych działającej po stronie klienta powstaje potencjalne niebezpieczeństwo przeprowadzenia ataku typu injection.
- **M8: podejmowanie ważnych decyzji na podstawie niezaufanych danych wejściowych**. Ryzyko w tej kategorii pojawia się, gdy decyzja dotycząca bezpieczeństwa jest podejmowana na podstawie danych wejściowych pochodzących z zaufanego źródła. W większości przypadków ryzyko wiąże się z mechanizmem komunikacji międzyprocesowej (IPC). Na przykład rozważ organizację posiadającą zestaw aplikacji komunikujących się z tym samym back endem. Programista decyduje, że zamiast wymagać uwierzytelnienia użytkownika w każdej aplikacji, będą one współdzieliły jeden token sesji. Takie podejście pozwala każdej aplikacji na uzyskanie dostępu do tokenu sesji. Do współdzielenia wspomnianego tokenu używany jest mechanizm IPC, taki jak dostawca treści. Jeżeli mechanizm IPC nie zostanie prawidłowo zabezpieczony, wszelkie inne zainstalowane w urządzeniu aplikacje o złośliwym działaniu mogą potencjalnie użyć interfejsu IPC w celu pobrania tokenu sesji i uzyskania dostępu do sesji użytkownika.
- **M9: nieprawidłowa obsługa sesji**. Zarządzanie sesją to ważna koncepcja podczas tworzenia oprogramowania. Sesja to mechanizm, który jest przez serwer używany do przechowywania informacji o stanie podczas pracy z protokołem bezstanowym, takim jak HTTP lub SOAP. Ryzyko tej kategorii obejmuje wszelkie luki w zabezpieczeniach powodujące ujawnienie tokenu sesji osobie atakującej. Pod pewnymi względami nakłada się na koncepcje przedstawione w punkcie „A2. Nieprawidłowe uwierzytelnianie i zarządzanie sesją" na liście[1] dziesięciu największych niebezpieczeństw dla aplikacji internetowych.

[1] Tę listę znajdziesz na stronie wiki pod adresem *https://www.owasp.org/index.php/Top_10_2013-Top_10 – przyp. tłum.*

- **M10: brak zabezpieczeń binarnych**. Ta kategoria ryzyka powoduje osłabienie mechanizmów obronnych, które programista może — i w większości przypadków powinien — wbudować w aplikację mobilną. Zabezpieczenia binarne z reguły spowalniają osobę atakującą, która próbuje analizować lub modyfikować kod binarny aplikacji albo przeprowadzić proces inżynierii odwrotnej.

Powyższa lista to niewątpliwie użyteczny zasób informacji o pojawiających się typach luk w zabezpieczeniach, które mogą wystąpić w aplikacjach mobilnych. Można się spodziewać, że wraz z postępującym coraz lepszym zabezpieczaniem aplikacji mobilnych będzie uzupełniana o nowo odkryte zagrożenia. Ponadto będzie odgrywała jeszcze większą rolę w edukacji programistów i profesjonalistów zajmujących się kwestiami zapewniania bezpieczeństwa.

Narzędzia rekomendowane przez OWASP Mobile Security

Narzędzia stanowią ważny element arsenału każdego profesjonalisty zajmującego się bezpieczeństwem, niezależnie od tego, czy są przeznaczone po prostu do wspomagania ręcznego przeprowadzania analizy, dostarczania frameworka do opracowania innych narzędzi czy pracy w charakterze zasobu zapewniającego możliwość wzmocnienia możliwości programistów. Projekt OWASP Mobile Security oferuje wiele narzędzi typu open source, które można wykorzystać podczas zapewniania bezpieczeństwa aplikacji. Informacje o nich znajdziesz na stronie wiki pod adresem *https://www.owasp.org/index.php/OWASP_Mobile_Security_Project#tab=Mobile_Tools*. Te narzędzia mogą okazać się użyteczne w trakcie poznawania tematu zapewniania bezpieczeństwa aplikacji mobilnych. Poniżej przedstawiłem krótkie omówienie wspomnianych narzędzi:

- **iMAS** (*https://www.owasp.org/index.php/OWASP_iMAS_iOS_Mobile_Application_*
  ↪*Security_Project*). Ten projekt utworzony przez MITRE Corporation jest rozprowadzanym jako *open source* frameworkiem bezpiecznej aplikacji dla systemu iOS. Stanowi idealny zasób dla programistów lub profesjonalistów zajmujących się zapewnianiem bezpieczeństwa, którzy chcą poznać lub zrozumieć implementację mechanizmów bezpieczeństwa na platformie iOS. Celem tego projektu jest zademonstrowanie i dostarczenie implementacji chroniących aplikacje iOS oraz dane w znacznie większym stopniu niż za pomocą modelu bezpieczeństwa oferowanego przez Apple. W efekcie powinno nastąpić wyraźne zmniejszenie ryzyka, że osoba atakująca będzie w stanie przeprowadzić proces inżynierii odwrotnej aplikacji, manipulować nią lub wykorzystać ewentualne luki w jej zabezpieczeniach. Aby osiągnąć postawiony sobie cel, w ramach projektu powstało wiele implementacji open source zapewniających rozwiązania w wielu obszarach, w których mogą występować luki w zabezpieczeniach, czyli między innymi w zakresie kodów PIN w aplikacji, wykrycia jailbreakingu urządzenia, ochrony przez debugowaniem i modyfikowaniem środowiska uruchomieniowego aplikacji. Choć niektórymi z wymienionych tematów dokładnie zajmiemy się w rozdziałach 2. i 3., projekt iMAS jest niewątpliwie użytecznym zasobem pomagającym w poznaniu technik defensywnych, a także może służyć w charakterze przewodnika dla programistów.

- **GoatDroid** (*https://www.owasp.org/index.php/Projects/OWASP_GoatDroid_Project*). Projekt GoatDroid, opracowany przez Jacka Mannino i Kena Johnsona, dostarcza samodzielne środowisko testowe dla aplikacji Android. To środowisko oferuje dwie przykładowe implementacje pomagające w doprowadzeniu do perfekcji własnych umiejętności. Pierwsza to FourGoats, czyli oparta na danych geolokalizacji sieć społecznościowa. Druga to Herd Financial, czyli fikcyjna aplikacja bankowości internetowej. Razem te dwa projekty zapewniają szerokie pokrycie większości niebezpieczeństw wymienionych wcześniej na liście OWASP Top 10 i stanowią dobry punkt wyjścia dla osób stawiających pierwsze kroki na obszarze zapewniania bezpieczeństwa aplikacji Android.
- **iGoat** (*https://www.owasp.org/index.php/OWASP_iGoat_Tool_Project*). Podobnie jak w przypadku projektu GoatDroid, iGoat to aplikacja treningowa pozwalająca Ci na udoskonalenie własnych umiejętności w zakresie oceny i analizy programów przeznaczonych dla platformy iOS. Ten projekt typu open source został opracowany przez Kena van Wyka, Jonathana Cartera i Seana Eidermillera. Kod źródłowy znajdziesz na stronie *https://code.google.com/archive/p/owasp-igoat/*. Dostarczony jest zarówno serwer, jak i aplikacja klienta wraz z wieloma ćwiczeniami dotyczącymi ważnych tematów, na przykład lokalnego magazynu danych, pęku kluczy, SQL injection itd.
- **Damn Vulnerable iOS** (*https://www.owasp.org/index.php/OWASP_DVIA*). Ten utworzony przez Prateeka Gianchandaniego projekt dostarcza kolejną zawierającą luki w zabezpieczeniach aplikację przeznaczoną do celów testowych. W połączeniu z projektem iGoat obie aplikacje zapewniają dość dobre pokrycie niebezpieczeństw wymienionych wcześniej na liście OWASP Top 10. Aplikacja składa się z wielu zadań przeznaczonych do wykonania, co pozwala na jeszcze dokładniejsze poznanie i zrozumienie tematów z zakresu bezpieczeństwa na platformie iOS. Uwzględnione zostały tutaj tematy pominięte w iGoat, na przykład wykrycie poddania urządzenia jailbreakingowi, manipulacje środowiskiem uruchomieniowym aplikacji, modyfikacje plików binarnych aplikacji, a także zagadnienia związane z kryptografią.
- **MobiSec** (*https://www.owasp.org/index.php/OWASP_Mobile_Security_Project*). To jest utworzone przez Tony'ego DeLaGrange'a i Kevina Johnsona środowisko typu live, przeznaczone do przeprowadzania testów penetracyjnych aplikacji mobilnych. Celem przyświecającym temu projektowi jest dostarczenie pojedynczego zasobu oraz utrzymywanie najnowszych wersji wszystkich narzędzi, które mogą okazać się przydatne podczas oceny i analizy aplikacji mobilnych. W ramach tego projektu otrzymujemy rozwiązanie podobne, jak stosowane w przypadku innych dystrybucji typu live, na przykład w popularnej Kali Linux. Jednak projekt MobiSec jest zorientowany na zapewnianie bezpieczeństwa na platformach mobilnych.
- **Androick** (*https://www.owasp.org/index.php/Projects/OWASP_Androick_Project*). W odróżnieniu od pozostałych projektów, ten dotyczy nieco innego tematu i jest skoncentrowany na zautomatyzowanych zadaniach analizy śledczej aplikacji na platformie Android, a nie na testach penetracyjnych lub samodzielnym udoskonalaniu umiejętności. Ten projekt, utworzony przez Floriana Pradinesa, automatyzuje zdobywanie kluczowych informacji i danych, jak pliki w formacie APK (ang. *android package*), dane aplikacji, bazy danych i dzienniki zdarzeń z urządzenia.

Oczywiście w trakcie swojej działalności na polu zapewniania bezpieczeństwa aplikacjom mobilnym napotkasz również inne, niewymienione tutaj narzędzia i nawet będziesz zmuszony do ich użycia. O wielu z nich wspomnę w dalszych rozdziałach. Jednak projekty przygotowane i zalecane przez OWASP są szczególnie użyteczne podczas doskonalenia własnych umiejętności, ponieważ zostały dobrze udokumentowane, są dostępne jako oprogramowanie open source oraz zostały opracowane specjalnie w celu pokrycia zagrożeń wymienionych na liście Top 10. Dlatego też gorąco zalecam zapoznanie się z tymi projektami i potraktowanie ich przez początkujących jako punktu wyjścia do dalszych eksperymentów.

Przyszłość dziedziny zapewniania bezpieczeństwa aplikacjom mobilnym

W ciągu minionych pięciu lat smartfony i aplikacje mobilne zyskały wielką popularność wśród użytkowników. Obecnie nie ma absolutnie żadnych znaków wskazujących na zahamowanie tego procesu i można oczekiwać, że ten trend będzie kontynuowany w przyszłości. Konsekwencją tej rosnącej rewolucji mobilnej jest coraz większy nacisk na poznanie i zrozumienie zagrożeń pojawiających się podczas wdrażania aplikacji mobilnych, a także opracowanie znacznie efektywniejszych sposobów ich pokonywania. Jestem przekonany, że obecne zagrożenia na polu bezpieczeństwa mobilnego nie są dobrze rozumiane, zwłaszcza w społeczności programistów.

Dlatego też można się spodziewać, że klasyczne luki w zabezpieczeniach, takie jak niezabezpieczony magazyn danych i niewystarczający poziom bezpieczeństwa warstwy transportowej, nadal będą dominowały w najbliższej przyszłości.

Z tego względu zapewnianie bezpieczeństwa na platformach mobilnych to nieustannie ewoluująca dziedzina i możemy się spodziewać pojawienia się nowych kategorii ataków wraz z kolejnymi zmianami w technologiach mobilnych. Wprowadzanie nowych komponentów sprzętowych, takich jak czytniki linii papilarnych, oraz coraz szersze wykorzystywanie istniejących technologii, na przykład NFC, bez wątpienia doprowadzi do odkrycia nowych luk w zabezpieczeniach. Szczególnie będzie to dotyczyć wdrożeń w środowiskach takich jak przetwarzanie płatności mobilnych, na przykład używanych przez technologie Google Wallet i Apple Pay.

Podobnie jak w przypadku innych obszarów oprogramowania, w szczególności używanych do obsługi transakcji finansowych, przestępcy będą szukali możliwości wykorzystania luk w zabezpieczeniach i łatwego zdobycia pieniędzy. Jesteśmy już świadkami wzrostu ilości oprogramowania typu malware atakującego aplikacje przeznaczone do obsługi bankowości internetowej oraz oszustw związanych z wiadomościami SMS typu premium. Należy spodziewać się kontynuacji tego trendu w przyszłości. Wspomniany wzrost doprowadził już do pewnych zmian w krajobrazie zagrożeń. W odpowiedzi na ten wzrost niektórzy programiści zaczęli stosować zabezpieczenia binarne, aby móc chronić aplikacje przed niebezpieczeństwami. Coraz większa świadomość czyhających niebezpieczeństw prawdopodobnie doprowadzi do szerszego stosowania tego rodzaju zabezpieczeń oraz użycia technologii takich jak dwupoziomowe uwierzytelnianie (ang. *two-factor authentication*).

Istnieje również prawdopodobieństwo dalszej ewolucji aplikacji mobilnych niezależnych od platformy sprzętowej, ponieważ programiści dążą do zmniejszenia fragmentacji wśród różnych platform mobilnych. To można dostrzec we wzroście dwóch wymienionych poniżej trendów w programowaniu:

- **Aplikacje oparte na przeglądarce WWW**. To pojęcie oznacza aplikacje, które są zwykle „przyjaznym urządzeniom mobilnym" klonem głównej witryny internetowej, wczytywanym poprzez przeglądarkę WWW urządzenia.
- **Aplikacje hybrydowe**. To pojęcie odwołuje się do aplikacji mobilnych będących natywnymi opakowaniami dla widoku sieciowego i bardzo często używających frameworka w celu uzyskania dostępu do natywnych funkcji urządzenia.

Uzupełnieniem dla wspomnianych trendów jest opracowanie ogromnej liczby zarówno komercyjnych, jak i dostępnych bezpłatnie frameworków, z których każdy ma swoje dziwactwa i niuanse prowadzące do powstania różnorodnych luk w zabezpieczeniach. Jak w przypadku większości zmian w technologii, te trendy przyniosły nowe rodzaje ataków oraz kolejne warianty istniejących. Związane z tym implikacje w obszarze zapewniania bezpieczeństwa przedstawię w rozdziale 14.

Pomimo wszystkich zmian w aplikacjach mobilnych nie ma żadnego znaku, że klasyczne ataki mogą zaniknąć. Jednak za pozytywny objaw można uznać coraz większą świadomość zagrożeń na polu bezpieczeństwa mobilnego i istniejących tutaj luk w zabezpieczeniach, co zawdzięczamy dokładnej dokumentacji, klasyfikacji oraz projektom takim jak opracowane przez grupę OWASP. Jestem przekonany, że poprzez wymienione i podobne projekty świadomość w zakresie bezpieczeństwa mobilnego będzie się zwiększała i pomoże w podjęciu przez programistów działań prowadzących do zmniejszenia liczby luk w zabezpieczeniach aplikacji mobilnych.

Podsumowanie

W ciągu minionych pięciu lat niezwykła popularność nowoczesnych smartfonów doprowadziła do gwałtownego skoku w dziedzinie tworzenia aplikacji mobilnych przez firmy zewnętrzne. Usprawnienia wprowadzone w smartfonach pomogły w szybkiej ewolucji aplikacji od prostych do oferujących potężne możliwości i zintegrowanych z wieloma technologiami internetowymi. Podczas tej ewolucji wiele aspektów technicznych, ekonomicznych i związanych z programowaniem miało wpływ na stosowanie kiepskiego poziomu zabezpieczeń, co można dostrzec w licznych obecnych aplikacjach mobilnych.

Poza tradycyjnymi problemami z zakresu bezpieczeństwa (dane wejściowe) wpływającymi na wszystkie rodzaje aplikacji, w aplikacjach mobilnych pojawiło się także wiele względnie unikatowych luk w zabezpieczeniach, co wynika z natury tych aplikacji. Te kwestie bardzo często nie są zbyt dobrze rozumiane przez programistów, co może prowadzić do ataków, gdy urządzenie jest używane w niezaufanej sieci, zostanie zgubione lub skradzione. Może nawet dojść do ataków przeprowadzanych przez inne komponenty platformy mobilnej.

Badania nad obecnym stanem bezpieczeństwa mobilnego pokazały, że luki w zabezpieczeniach aplikacji nie są dobrze rozumiane, a większość aplikacji pozostaje podatna na ataki. Co więcej, ewolucja nowych technologii oraz kolejne integracje prawdopodobnie doprowadzą do powstania zupełnie nowych kategorii ataków, które będą mogły narazić organizacje na poważne niebezpieczeństwo, o ile w porę nie zostaną podjęte odpowiednie działania.

Skorowidz

B

C

D

E

F

G

H

I

L

N

O

P

R

S

Ś

T

U

V

W

X

Z

Ż

Aplikacja mobilna
— popatrz na nią oczami hakera i zabezpiecz ją!

Urządzenia mobilne zapewniają ogromną wygodę. Natychmiastowy dostęp do informacji czy dokumentu, niezależnie od lokalizacji użytkownika, jest czymś oczywistym. W ten sposób wszelkie ważne i wrażliwe informacje, takie jak dane pozwalające na identyfikację, dane finansowe czy poufne dokumenty, są cały czas na wyciągnięcie ręki — niestety, często ta ręka należy do kogoś, kto w żadnym razie nie powinien takich informacji uzyskać. Każdy, kto pisze aplikacje mobilne, musi pamiętać o kwestiach związanych z ich bezpieczeństwem. Konsekwencje nieuprawnionego dostępu do danych mogą być niezwykle poważne!

Ta książka jest całościowym, a równocześnie bardzo praktycznym kompendium wiedzy o bezpieczeństwie aplikacji mobilnych. Uwzględniono tu problemy charakterystyczne dla platform iOS, Android i Windows Phone, dzięki czemu zaproponowanie najwłaściwszej strategii zabezpieczenia aplikacji jest o wiele prostsze. Wyjaśniono przyczyny podatności aplikacji mobilnych na ataki, opisano też techniki prowadzenia ataku i wykorzystywania luk w zabezpieczeniach. Bardzo dokładnie przedstawiono także strategie obrony i działania, dzięki którym programiści mogą chronić swoje aplikacje. Poradnik ten docenią przede wszystkim osoby przeprowadzające testy penetracyjne, konsultanci z zakresu bezpieczeństwa oraz oczywiście programiści.

Najciekawsze zagadnienia:

- 10 najważniejszych zagrożeń aplikacji mobilnych według OWASP Mobile Security
- Analiza aplikacji i identyfikowanie problemów bezpieczeństwa
- Ataki typu injection, brute force, XSS, tapjacking i wiele innych
- Wykorzystanie inżynierii wstecznej
- Mechanizmy obronne w aplikacjach na poszczególne platformy
- Zabezpieczanie aplikacji niezależnych od platformy

DOMINIC CHELL
jest ekspertem w dziedzinie zabezpieczeń aplikacji na platformę iOS. Jest znany z wystąpień na konferencjach oraz jako autor wielu publikacji dotyczących bezpieczeństwa urządzeń mobilnych.

TYRONE ERASMUS
zajmuje się badaniami z zakresu bezpieczeństwa systemów. Interesuje się testami penetracyjnymi. Bada nowe narzędzia i techniki pojawiające się w tej sferze.

SHAUN COLLEY
jest ekspertem z zakresu bezpieczeństwa w urządzeniach mobilnych i oceny kodu natywnego. Zajmuje się też inżynierią wsteczną.

OLLIE WHITEHOUSE
specjalizuje się w zapewnianiu bezpieczeństwa aplikacji i urządzeń mobilnych, interesuje się też technikami bezprzewodowymi.

Bardzo dobre, szerokie wprowadzenie do tematyki bezpieczeństwa aplikacji mobilnych. Pokazuje nie tylko jak zabezpieczać, ale również jak testować aplikacje pod kątem ataków. Brak wypełniaczy i struktura sprawia, że każdy ma szansę zdobyć wiedzę w sposób rzeczywiście efektywny. Stanowi unikalną mapę, która bardzo często wspiera Czytelnika w wychodzeniu również poza omówione obszary.

MATEUSZ BILIŃSKI
HEAD OF MOBILE SECURITY
W NIEBEZPIECZNIK.PL

Helion
helion.pl
HELION SA
ul. Kościuszki 1c
44-100 Gliwice
tel.: 32 230 98 63
helion@helion.pl

KOD KORZYŚCI
Sięgnij po więcej!

ISBN 978-83-8322-548-7

Cena: 129,00 zł