INTRODUÇÃO AO PENTEST EM APLICAÇÕES MOBILE PT-1

JOAS ANTONIO

AUTOR

- Nome – Joas Antonio;
- Entusiasta e apaixonado por segurança da informação;

EBOOK

- Esse e-book tem como objetivo trazer conceitos básicos em pentest mobile;
- É um guia de estudos para se aprofundar nessa área;
- O livro é bem básico e feito para iniciantes na área;
- Contém um apanhado de links com materiais de estudos e técnicas práticas que eu salvei e estou compartilhando com vocês;

CONCEITOS DE PENTEST EM APP MOBILE

INTRODUÇÃO MAPTM

- A Metodologia de Teste de Penetração de Aplicativos Móveis (MAPTM), conforme descrito pelo autor Vijay Kumar Velu em seu ebook , é o procedimento que deve ser seguido durante a realização do pentest nos aplicativos mobile. Ele é baseado na metodologia de segurança de aplicativos e muda o foco do AppSec tradicional, que considera a ameaça primária como originada da Internet.

INTRODUÇÃO MAPTM

O MAPTM é dividido em quatro etapas:

- A descoberta: requer que o pentester colete informações essenciais para a compreensão de eventos que levam à exploração bem-sucedida de aplicativos móveis.
- A avaliação ou análise: envolve o pentester examinar o código-fonte do aplicativo mobile e identificar os pontos fracos que podem ser explorados.
- A exploração envolve o pentester aproveitar as vulnerabilidades descobertas para tirar vantagem do aplicativo de uma maneira não pretendida pelo programador inicialmente.
- O relatório é a etapa final da metodologia e envolve registrar e apresentar os problemas descobertos de uma maneira que faça sentido para a gestão. Este também é o estágio que diferencia um pentest de um ataque. Segue-se uma discussão mais detalhada das quatro etapas.

MAPTM - DESCOBERTA

1) Descoberta

A coleta de informação é a etapa mais importante em um pentest. A capacidade de descobrir pistas ocultas que podem lançar luz sobre a existência de uma vulnerabilidade pode ser a diferença entre um pentest bem-sucedido e malsucedido.

O processo de descoberta envolve

Open Source Intelligence (OSINT): O pentester pesquisa na Internet por informações sobre o aplicativo. Isso pode ser encontrado em mecanismos de pesquisa e sites de redes sociais, código-fonte vazado por meio de repositórios de código-fonte, fóruns de desenvolvedores ou mesmo na dark web.

Compreendendo a plataforma: é importante para o testador de penetração entender a plataforma de aplicativo móvel, mesmo de um ponto de vista externo, para ajudar no desenvolvimento de um modelo de ameaça para o aplicativo. O pentester leva em consideração a empresa por trás do aplicativo, seu caso de negócios e partes interessadas relacionadas. As estruturas e processos internos também são levados em consideração.

MAPTM - DESCOBERTA

Cenários do lado do cliente versus cenários do lado do servidor: O PenTester precisa ser capaz de entender o tipo de aplicativo (nativo, híbrido ou da web) e trabalhar nos casos de teste. As interfaces de rede do aplicativo, dados do usuário, comunicação com outros recursos, gerenciamento de sessão, comportamento de desbloqueio / root são todos considerados aqui. Considerações de segurança também são feitas; por exemplo, o aplicativo interage com firewalls? Bancos de dados ou servidores? Quão seguro é isso?

As informações coletadas podem incluir:

- **A sessão do usuário permanece ativa até que um logoff manual seja executado.**
- **Nenhuma transação financeira é realizada.**
- **O aplicativo foi desenvolvido para não funcionar em dispositivos desbloqueados.**
- **As ações executadas no servidor incluem adições, exclusões e pulls do banco de dados.**

MAPTM - ANALISE

2) Avaliação / Análise

O processo de avaliação de aplicativos móveis é único porque requer que o testador de penetração verifique os aplicativos antes e depois da instalação. As diferentes técnicas de avaliação que são encontradas no MAPTM incluem:

Análise de arquivo local— O pentester verifica os arquivos locais gravados no sistema de arquivos pelo aplicativo para garantir que não haja violações.

Análise de arquivo— O testador de penetração extrai os pacotes de instalação do aplicativo para as plataformas Android e iOS. Uma revisão é feita para garantir que não haja modificações feitas nas configurações do binário compilado.

MAPTM - ANALISE

Engenharia reversa - envolve a conversão dos aplicativos compilados em código-fonte legível. O testador de penetração analisa o código legível para entender a funcionalidade interna do aplicativo e procurar vulnerabilidades. O código-fonte do aplicativo Android pode ser modificado depois de revertido e recompilado. As seguintes ferramentas podem ser usadas durante a realização de engenharia reversa:

- **Android — dex2jar, JD-GUI**
- **iOS — otool, class-dump-z**

Análise estática— Durante a análise estática, o testador de penetração não executa o aplicativo. A análise é feita nos arquivos fornecidos ou código-fonte descompilado.

MAPTM - ANALISE

Análise dinâmica— O pentester analisa o aplicativo móvel conforme ele é executado no dispositivo. As revisões feitas incluem análise forense do sistema de arquivos, avaliação do tráfego de rede entre o aplicativo e o servidor e uma avaliação da comunicação entre processos do aplicativo (IPC).

Existem algumas ferramentas disponíveis para o pentester para análise automática e manual do código-fonte. Esses incluem:

- **Android: Androwarn, Andrubis e ApkAnalyser**
- **iOS: Flawfinder e Clang Static Analyzer**

MAPTM - ANALISE

Análise de endpoint de comunicação entre processos : O pentester analisa os diferentes endpoints IPC de aplicativos móveis. A avaliação é realizada em:

- **Provedores de conteúdo - garantem que o acesso aos bancos de dados seja obtido.**
- **Intents - Esses são sinais usados para enviar mensagens entre componentes do sistema Android.**
- **Receptores de transmissão - recebem e atuam de acordo com as intenções recebidas de outros aplicativos no sistema Android.**
- **Atividades - Estas constituem as telas ou páginas do aplicativo.**
- **Serviços - são executados em segundo plano e realizam tarefas independentemente de o aplicativo principal estar em execução.**

MAPTM - ANALISE

As informações obtidas na avaliação podem ser usadas para criar um modelo de ameaça. Por exemplo, podemos considerar o seguinte:

- Vetor descoberto— O aplicativo se comunica com um banco de dados em um servidor remoto.
- Possível ameaça - Leitura não autorizada do tráfego de dados durante a comunicação com o servidor.
- Contramedidas relacionadas— Implementar uma proteção de camada de transporte seguro (SSL, TLS).
- Possível caso de teste - tentativa de detectar o tráfego entre o aplicativo e o back-end do servidor.

MAPTM - EXPLORAÇÃO

O pentester age com base nas informações descobertas no processo de coleta de informações para atacar o aplicativo móvel. A coleta de inteligência realizada de forma cuidadosa garante uma grande chance de exploração bem-sucedida, portanto, um projeto bem-sucedido.

O pentester tenta explorar a vulnerabilidade a fim de obter informações confidenciais ou realizar atividades maliciosas e, por fim, executa o escalonamento de privilégios para elevar ao usuário mais privilegiado (root), de modo a não enfrentar quaisquer restrições nas atividades realizadas.

O pentester então persiste dentro do dispositivo comprometido. Isso significa simplesmente que ele / ela executa módulos que permitem o backdoor do dispositivo com o objetivo de mostrar a capacidade de realizar acessos futuros.

MAPTM - EXPLORAÇÃO

3) Exploração

O pentester age com base nas informações descobertas no processo de coleta de informações para atacar o aplicativo móvel. A coleta de inteligência realizada de forma cuidadosa garante uma grande chance de exploração bem-sucedida, portanto, um projeto bem-sucedido.

O pentester tenta explorar a vulnerabilidade a fim de obter informações confidenciais ou realizar atividades maliciosas e, por fim, executa o escalonamento de privilégios para elevar ao usuário mais privilegiado (root), de modo a não enfrentar quaisquer restrições nas atividades realizadas.

O pentester então persiste dentro do dispositivo comprometido. Isso significa simplesmente que ele / ela executa módulos que permitem o backdoor do dispositivo com o objetivo de mostrar a capacidade de realizar acessos futuros.

MAPTM - RELATÓRIO

4) Relatórios

Um bom relatório se comunica com a gerência em linguagem simples, indicando claramente as vulnerabilidades descobertas, consequências para o negócio e possíveis correções ou recomendações.

As vulnerabilidades devem ter classificação de risco e comunicação técnica adequada feita para o pessoal técnico, com uma prova de conceito incluída para apoiar as descobertas descobertas.

https://resources.infosecinstitute.com/introduction-mobile-application-penetration-testing-methodology/

OWASP TOP 10 MOBILE SECURITY

M1: Improper Platform Usage (https://owasp.org/www-project-mobile-top-10/2016-risks/m1-improper-platform-usage)

M2: Insecure Data Storage (https://owasp.org/www-project-mobile-top-10/2016-risks/m2-insecure-data-storage)

M3: Insecure Communication (https://owasp.org/www-project-mobile-top-10/2016-risks/m3-insecure-communication)

M4: Insecure Authentication (https://owasp.org/www-project-mobile-top-10/2016-risks/m4-insecure-authentication)

M5: Insufficient Cryptography (https://owasp.org/www-project-mobile-top-10/2016-risks/m5-insufficient-cryptography)

M6: Insecure Authorization (https://owasp.org/www-project-mobile-top-10/2016-risks/m6-insecure-authorization)

M7: Client Code Quality (https://owasp.org/www-project-mobile-top-10/2016-risks/m7-client-code-quality)

M8: Code Tampering (https://owasp.org/www-project-mobile-top-10/2016-risks/m8-code-tampering)

M9: Reverse Engineering (https://owasp.org/www-project-mobile-top-10/2016-risks/m9-reverse-engineering)

M10: Extraneous Functionality (https://owasp.org/www-project-mobile-top-10/2016-risks/m10-extraneous-functionality)

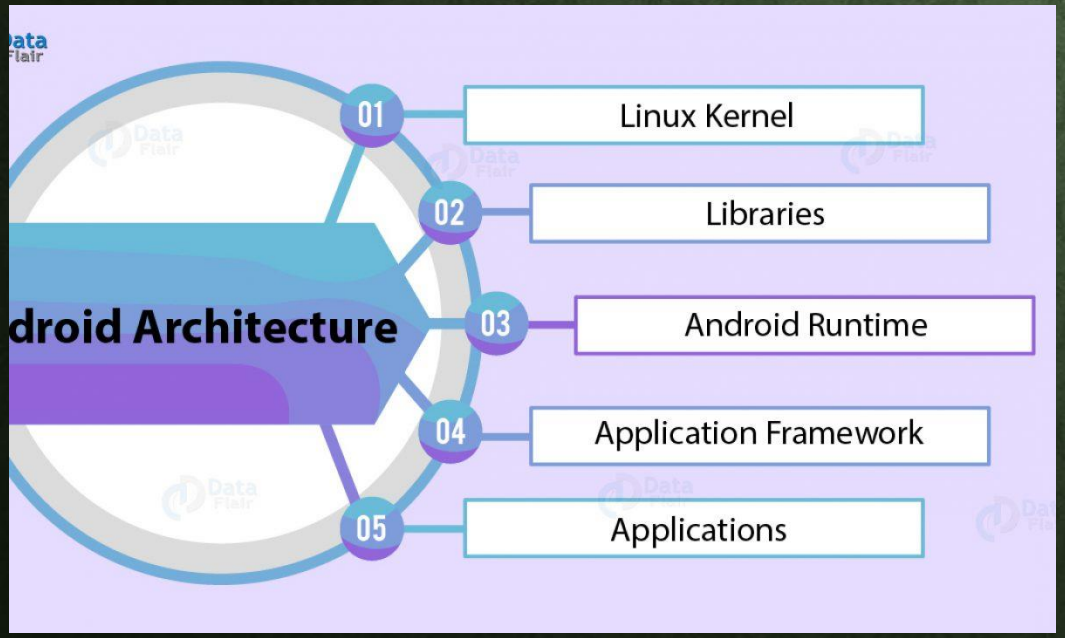
ANDROID ARQUITETURA

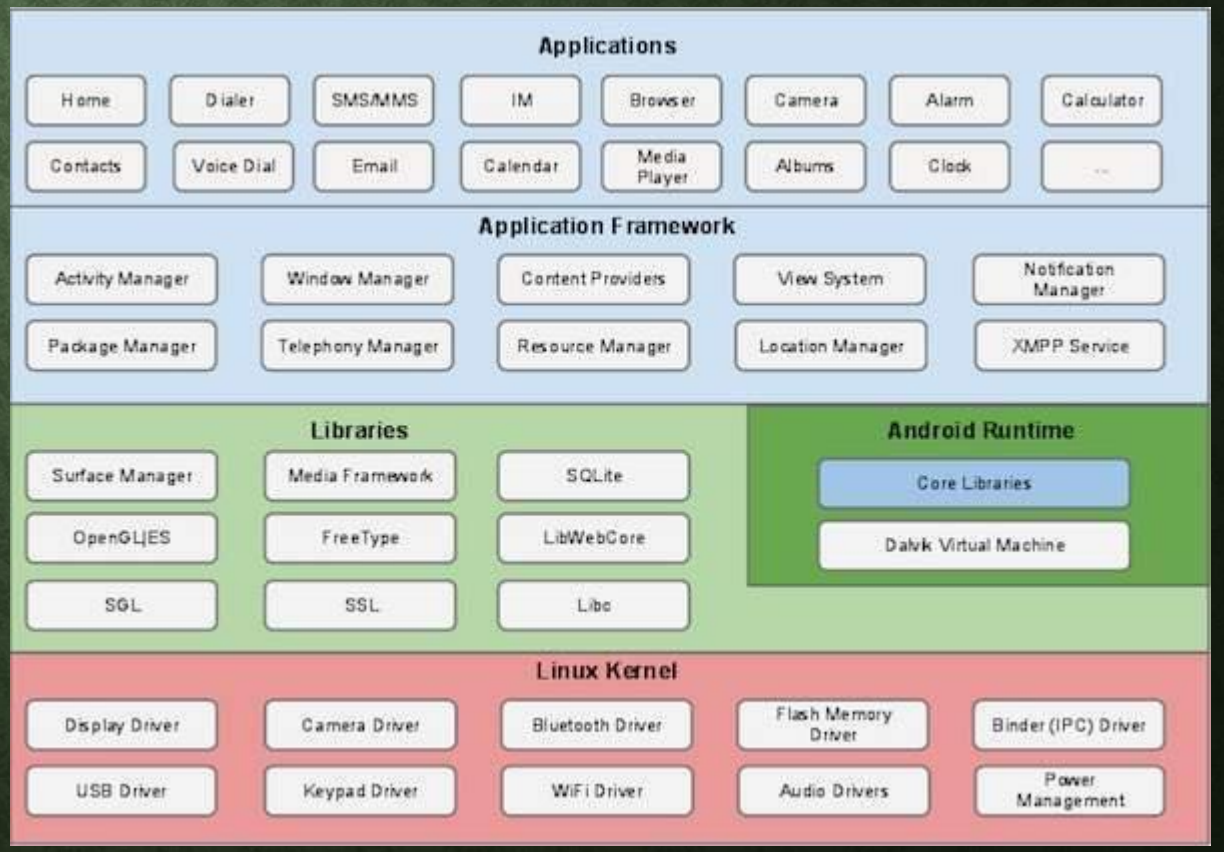
https://developer.android.com/topic/libraries/architecture

https://developer.android.com/jetpack/guide

https://developer.android.com/guide/platform

https://www.tutorialspoint.com/android/android_architecture.htm





ANDROID ARQUITETURA

https://www.youtube.com/watch?v=mnC1aN2yhqI&ab_channel=NelsonGlauber

https://www.youtube.com/watch?v=bTblJ617PV4&ab_channel=iMasters

https://medium.com/@paulo_linhares/android-room-android-architecture-components-arc-ba44a6640e7c

https://arctouch.com/blog/android-architecture-components-jetpack/

https://www.youtube.com/watch?v=t92M0a5nBIw

https://www.youtube.com/watch?v=h2AwlhBRooM

IOS ARQUITETURA

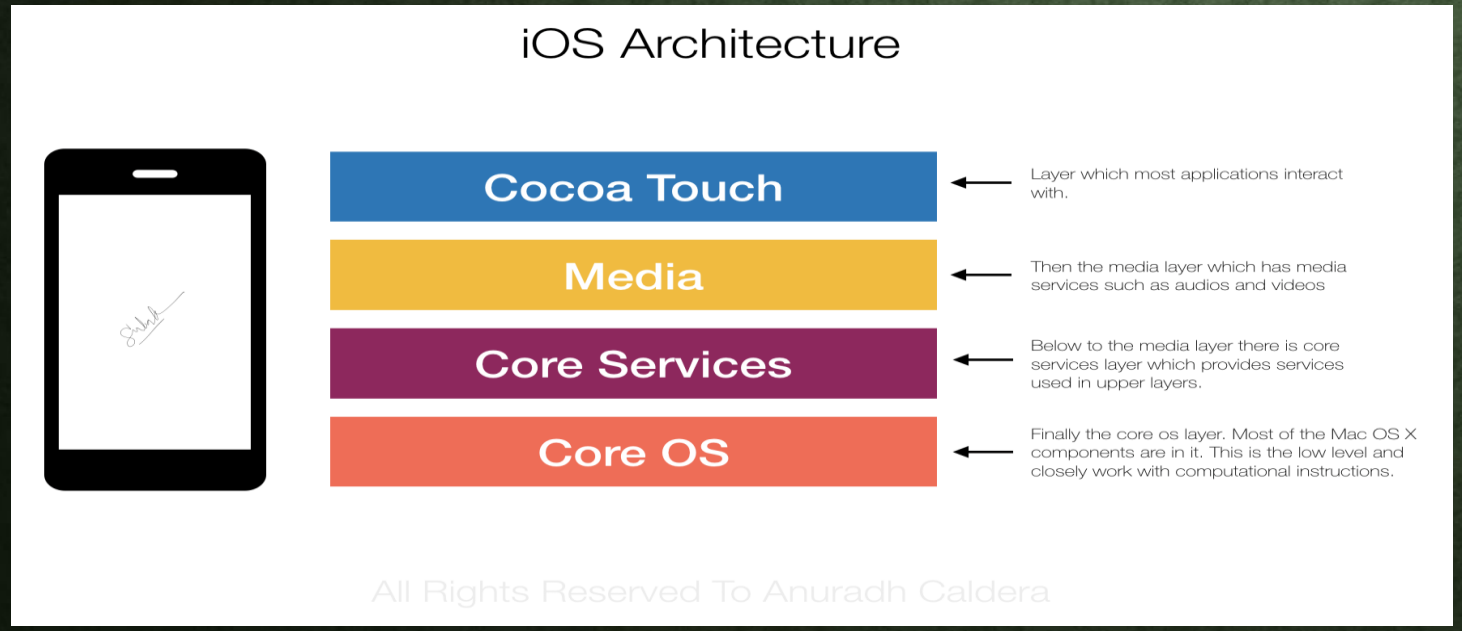
https://medium.com/@anuradhs/ios-architecture-a2169dad8067

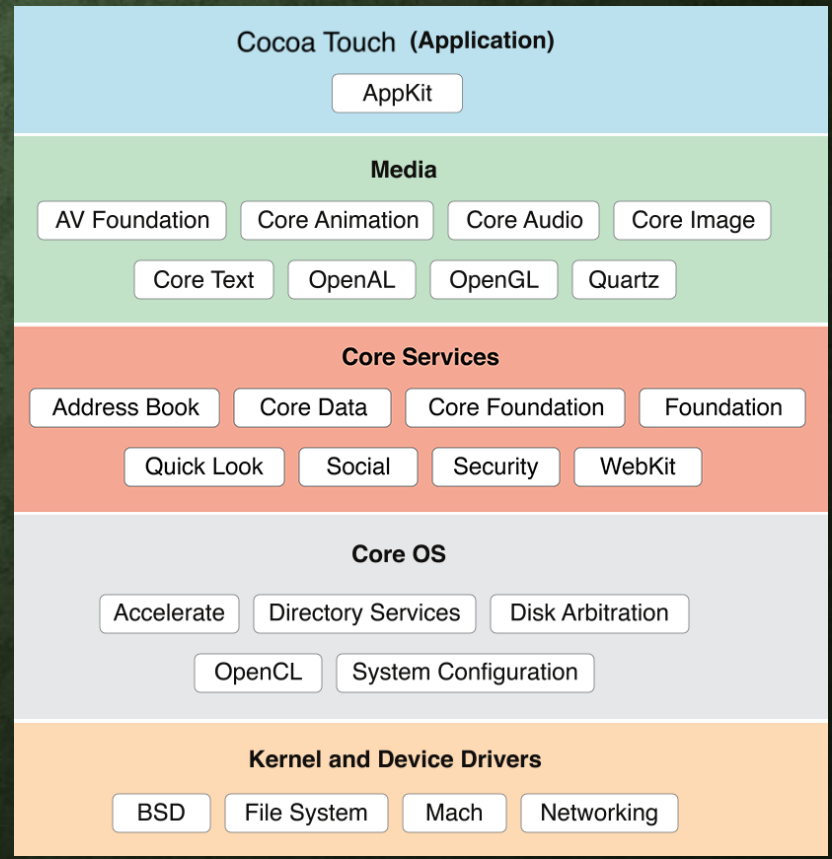
https://www.tutorialspoint.com/apple-ios-architecture#:~:text=The%20iOS%20architecture%20is%20layered,user%20interface%20and%20sophisticated%20graphics.

https://medium.com/ios-os-x-development/ios-architecture-patterns-ecba4c38de52

https://developer.apple.com/design/human-interface-guidelines/ios/app-architecture/launching/

https://intellipaat.com/blog/tutorial/ios-tutorial/ios-architecture/





IOS ARQUITETURA

https://www.youtube.com/watch?v=R1GWWONkJvU&ab_channel=IGTI

https://www.youtube.com/watch?v=R1GWWONkJvU&ab_channel=IGTI

https://www.youtube.com/watch?v=W_SNdRz0cd8&ab_channel=Viralizou

https://www.youtube.com/watch?v=pez7eBgQi60&ab_channel=DOitSA

PENTEST EM APP MOBILE – PRINCIPAIS CONCEITOS

https://medium.com/netsentries/mobile-application-penetration-testing-784499d9611c

https://blog.rsisecurity.com/what-you-need-to-know-about-mobile-penetration-testing/

https://mobile-security.gitbook.io/mobile-security-testing-guide/overview/0x04b-mobile-app-security-testing

https://www.tutorialspoint.com/mobile_security/mobile_security_pen_testing.htm

PENTEST EM APP MOBILE - GUIA

ENGENHARIA REVERSA – ESTÁTICA E DINÂMICA

- https://god.owasp.de/archive/2018/slides/2018-god-holguera.pdf
- https://techbeacon.com/app-dev-testing/how-hack-app-8-best-practices-pen-testing-mobile-apps
- https://medium.com/@chris.yn.chen/apk-reverse-engineering-df7ed8cec191
- https://medium.com/swlh/reverse-engineering-and-modifying-an-android-game-apk-ctf-c617151b874c
- https://www.youtube.com/watch?v=VLsJETb3m4M
- https://www.youtube.com/watch?v=7SRfk321I5o&ab_channel=RSAConference
- https://www.youtube.com/watch?v=ZLDGtSN_m38&ab_channel=MahmudAhsan
- https://www.youtube.com/watch?v=eHdDS2e_qf0&list=PL4zZ9lJ-RCbfv6f6Jc8cJ4ljKqENkTfi7&ab_channel=HackN%27RollAcademy
- https://www.secplicity.org/2019/09/11/android-apk-reverse-engineering-whats-in-an-apk/
- https://ragingrock.com/AndroidAppRE/app_fundamentals.html
- https://github.com/tanprathan/MobileApp-Pentest-Cheatsheet
- http://mobiletools.mwrinfosecurity.com/Using-Drozer-for-application-security-assessments/
- https://pentestlab.blog/2017/02/06/reverse-engineering-android-applications/
- https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05c-Reverse-Engineering-and-Tampering.md
- https://medium.com/@thomas_shone/reverse-engineering-apis-from-android-apps-part-1-ea3d07b2a6c
- https://www.rsaconference.com/writable/presentations/file_upload/stu-w02b-beginners-guide-to-reverse-engineering-android-apps.pdf
- https://resources.infosecinstitute.com/android-hacking-security-part-6-exploiting-debuggable-android-applications/#gref

JAIL BREAK

Jail Break Detection Bypass

- https://www.notsosecure.com/bypassing-jailbreak-detection-ios/
- https://www.theiphonewiki.com/wiki/Bypassing_Jailbreak_Detection
- https://resources.infosecinstitute.com/ios-application-security-part-44-bypassing-jailbreak-detection-using-xcon/#gref
- https://blog.attify.com/bypass-jailbreak-detection-frida-ios-applications/
- https://www.c0d3xpl0it.com/2017/05/ios-jailbreak-bypass-using-needle.html
- https://resources.infosecinstitute.com/ios-application-security-part-23-jailbreak-detection-evasion/
- https://agostini.tech/2018/02/05/ios-application-security-part-three-bypassing-jailbreak-and-certificate-pinning-let-the-right-one-in/

JAIL BREAK

Cert Pinning Bypass

- https://blog.netspi.com/four-ways-to-bypass-ios-ssl-verification-and-certificate-pinning/
- https://www.nccgroup.trust/us/about-us/newsroom-and-events/blog/2015/january/bypassing-openssl-certificate-pinning-in-ios-apps/
- https://github.com/vtky/Swizzler2/wiki/Case-Study:-SSL-Pinning
- https://labs.nettitude.com/tutorials/using-frida-to-bypass-snapchats-certificate-pinning/

Static and Dynamic Analysis

- https://medium.com/@ansjdnakjdnajkd/dynamic-analysis-of-ios-apps-wo-jailbreak-1481ab3020d8
- https://labs.mwrinfosecurity.com/assets/BlogFiles/Needle-Finding-Issues-within-iOS-Applications.pdf
- https://medium.com/@drag0n/needle-analysis-of-ios-mobile-applications-cfd9e407c0d9

JAIL BREAK

Reverse Engineering

- https://labs.mwrinfosecurity.com/blog/repacking-and-resigning-ios-applications/
- https://github.com/OWASP/owasp-mstg/blob/master/Document/0x06c-Reverse-Engineering-and-Tampering.md
- https://resources.infosecinstitute.com/ios-application-security-part-2-getting-class-information-of-ios-apps/
- https://resources.infosecinstitute.com/penetration-testing-for-iphone-applications-part-5

Misc

- https://www.igeeksblog.com/how-to-sideload-apps-on-iphone-ipad-in-ios-10/

MOBILE PENTEST TOOLS

- https://project-awesome.org/ashishb/android-security-awesome
- https://www.softwaretestinghelp.com/mobile-app-security-testing-tools/
- https://www.softwaretestinghelp.com/mobile-app-pen-testing-tools-service-providers/
- https://resources.infosecinstitute.com/top-6-mobile-application-penetration-testing-tools/
- https://mobile-security.gitbook.io/mobile-security-testing-guide/appendix/0x08-testing-tools
- https://github.com/georgiaw/Smartphone-Pentest-Framework

SMARTPHONE PENTEST FRAMEWORK

- https://www.youtube.com/watch?v=lfh797dgoN0&ab_channel=AdrianCrenshaw
- https://www.youtube.com/watch?v=YEqL87eXLfU&ab_channel=BSidesLV
- https://www.youtube.com/watch?v=JygeotdtZtk&lc=UggVoDYIgc4qbXgCoAEC&ab_channel=OWASP

MOBILE TESTING BURP SUITE

- https://resources.infosecinstitute.com/pentesting-mobile-applications-burpsuite/
- https://portswigger.net/burp/documentation/desktop/mobile-testing
- https://www.bugcrowd.com/blog/mobile-testing-setting-up-your-android-device-part-1/
- https://medium.com/@Mayank.Grover/intercept-ssl-traffic-to-perform-penetration-testing-on-android-apps-using-charles-debug-proxy-59211859d22f
- https://mundohacker.net.br/android-ssl-pinning/
- https://medium.com/@appmattus/android-security-ssl-pinning-1db8acb6621e
- https://www.netguru.com/codestories/3-ways-how-to-implement-certificate-pinning-on-android
- https://github.com/wultra/ssl-pinning-android
- https://levelup.gitconnected.com/bypassing-ssl-pinning-on-android-3c82f5c51d86?gi=91eaa703c92f
- https://www.mcafee.com/enterprise/en-us/assets/misc/ms-bypass-ssl-pinning-android-4-6.pdf

INSECURE LOCAL STORAGE

- https://resources.infosecinstitute.com/android-hacking-security-part-9-insecure-local-storage-shared-preferences/
- https://medium.com/@andreacioccarelli/android-sharedpreferences-data-weakness-66a44f070e76
- https://www.areizen.fr/post/hackingsharedpreferences/
- https://www.exploit-db.com/exploits/44852
- https://manifestsecurity.com/android-application-security-part-8/

MEMORY CORRUPTION

- https://android-developers.googleblog.com/2020/02/detecting-memory-corruption-bugs-with-hwasan.html
- https://www.blackhat.com/docs/us-15/materials/us-15-Gong-Fuzzing-Android-System-Services-By-Binder-Call-To-Escalate-Privilege-wp.pdf
- https://vimeo.com/444169237

EXPLOIT EM COMPONENTES ANDROID

- https://securitygrind.com/exploiting-android-components-abusing-activities/
- https://bsderek.home.blog/2020/02/12/exploit-activity-component-in-insecurebankv2-application/
- https://resources.infosecinstitute.com/android-hacking-security-part-1-exploiting-securing-application-components/
- https://conference.hitb.org/hitbsecconf2011kul/materials/D1T1%20-%20Riley%20Hassell%20-%20Exploiting%20Androids%20for%20Fun%20and%20Profit.pdf

EXPLOIT EM COMPONENTES ANDROID

- https://securitygrind.com/exploiting-android-components-abusing-activities/
- https://bsderek.home.blog/2020/02/12/exploit-activity-component-in-insecurebankv2-application/
- https://resources.infosecinstitute.com/android-hacking-security-part-1-exploiting-securing-application-components/
- https://conference.hitb.org/hitbsecconf2011kul/materials/D1T1%20-%20Riley%20Hassell%20-%20Exploiting%20Androids%20for%20Fun%20and%20Profit.pdf

SMALI DEBUGGER

- https://living-sun.com/pt/android/10221-how-to-debug-smali-code-of-an-android-application-android-apk-dex-smali.html
- https://rafaelcintralopes.com.br/engenharia-reversa-em-aplicativos-android/
- https://malacupa.com/2018/11/11/debug-decompiled-smali-code-in-android-studio-3.2.html
- https://medium.com/@ghxst.dev/static-analysis-and-debugging-on-android-using-smalidea-jdwp-and-adb-b073e6b9ae48
- https://www.youtube.com/watch?v=krJ8w6drjv4
- https://www.youtube.com/watch?v=pn_CgHbl00E&ab_channel=SanjayGondaliya
- https://www.youtube.com/watch?v=uc7eZGE07ps&ab_channel=0xFFSweden

ADB

- https://gbhackers.com/android-application-penetration-test-part-3/
- https://blog.usejournal.com/an-intro-to-pentesting-an-android-phone-464ec4860f39
- https://github.com/mirfansulaiman/Command-Mobile-Penetration-Testing-Cheatsheet
- https://book.hacktricks.xyz/mobile-apps-pentesting/android-app-pentesting/adb-commands
- https://www.apriorit.com/dev-blog/654-reverse-pentesting-android-apps
- https://www.cin.ufpe.br/~tg/2017-2/dam4-tg.pdf
- https://medium.com/@0xklaue/android-penetration-testing-ba362e03d89e

IOS PENTEST

- https://resources.infosecinstitute.com/ios-application-security-part-1-setting-up-a-mobile-pentesting-platform/
- https://resources.infosecinstitute.com/pentesting-iphone-applications/
- https://medium.com/securing/pentesting-ios-apps-without-jailbreak-91809d23f64e
- https://payatu.com/6-must-tools-ios-pentesting-toolkit
- https://devcount.com/ios-pentesting-tools/
- https://research.nccgroup.com/wp-content/uploads/2020/07/introducing_idb_-_simplified_blackbox_ios_app_pentesting.pdf

APK PAYLOADS

- https://www.hackingloops.com/android-penetration-testing-using-metasploit-framework/
- https://resources.infosecinstitute.com/lab-hacking-an-android-device-with-msfvenom/
- https://resources.infosecinstitute.com/lab-android-exploitation-with-kali/
- https://null-byte.wonderhowto.com/how-to/embed-metasploit-payload-original-apk-file-0166901/
- https://gist.github.com/davidlares/6af9541c14d7a684763b44a5e33d9193
- https://medium.com/@irfaanshakeel/hacking-android-phone-remotely-using-metasploit-43ccf0fbe9b8
- https://medium.com/@IamLucif3r/hacking-android-device-remotely-40800e30813
- https://hackersploit.org/android-hacking-tutorials/
- https://www.hackingarticles.in/android-mobile-exploitation-evil-droid/
- https://medium.com/@chamo.wijetunga/how-to-hack-into-android-with-an-injected-malicious-application-adad98f6a80b

MALWARE APK DEVELOPER

- https://null-byte.wonderhowto.com/how-to/make-your-malicious-android-app-be-more-convincing-0163730/
- https://www.sciencedirect.com/topics/computer-science/malware-developer
- https://pt.slideshare.net/schenette/building-custom-android-malware-brucon-2013
- https://github.com/geeksonsecurity/android-overlay-malware-example
- https://github.com/soarlab/maline
- https://null-byte.wonderhowto.com/how-to/embed-metasploit-payload-original-apk-file-part-2-do-manually-0167124/
- https://www.insiderattack.net/2013/09/android-malware-injection-into-original_5.html
- https://resources.infosecinstitute.com/top-7-android-ransomware-threats/

MALWARE APK DEVELOPER

- https://null-byte.wonderhowto.com/how-to/make-your-malicious-android-app-be-more-convincing-0163730/
- https://www.sciencedirect.com/topics/computer-science/malware-developer
- https://pt.slideshare.net/schenette/building-custom-android-malware-brucon-2013
- https://github.com/geeksonsecurity/android-overlay-malware-example
- https://github.com/soarlab/maline
- https://null-byte.wonderhowto.com/how-to/embed-metasploit-payload-original-apk-file-part-2-do-manually-0167124/
- https://www.insiderattack.net/2013/09/android-malware-injection-into-original_5.html
- https://resources.infosecinstitute.com/top-7-android-ransomware-threats/

ANDROID BYPASS RESTRICTION

- https://androidreverse.wordpress.com/2020/05/02/android-api-restriction-bypass-for-all-android-versions/
- https://github.com/ChickenHook/RestrictionBypass
- https://stackoverflow.com/questions/55970137/bypass-androids-hidden-api-restrictions
- https://blog.quarkslab.com/android-runtime-restrictions-bypass.html
- https://github.com/quarkslab/android-restriction-bypass
- https://medium.com/@7anac/bypass-restricted-or-blocked-screenshots-on-android-2020-46871f1b2271
- https://www.slashgear.com/thousands-of-android-apps-bypass-permissions-to-violate-user-privacy-09583325/
- https://www.reddit.com/r/GrapheneOS/comments/es1ado/android_11_will_harden_hidden_api_restrictions/

MALICIOUS INTENTS ANDROID APK

- https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/compromising-android-applications-with-intent-manipulation/
- https://www.cs.rice.edu/~vs3/PDF/PermissionFlow-TR.pdf
- https://cwe.mitre.org/data/definitions/926.html
- https://community.veracode.com/s/question/0D52T00004rDtkF/how-can-we-avoid-improper-export-of-android-application-components
- https://androidforums.com/threads/improper-export-of-android-application-components.1315110/
- https://securityintelligence.com/new-vulnerability-android-framework-fragment-injection/

CWE MOBILE

- https://cwe.mitre.org/data/definitions/919.html
- https://static.googleusercontent.com/media/www.google.com/pt-BR//about/appsecurity/play-rewards/Android_app_vulnerability_classes.pdf

DAMN VULNERABILITY IOS

- http://damnvulnerableiosapp.com/
- https://github.com/prateek147/DVIA-v2
- https://github.com/prateek147/DVIA
- https://blog.attify.com/tag/damn-vulnerable-ios-app/
- https://resources.infosecinstitute.com/ios-application-security-part-45-enhancements-in-damn-vulnerable-ios-app-version-2-0/
- https://resources.infosecinstitute.com/getting-started-damn-vulnerable-ios-application/

EXPLORING FILESYSTEM IOS

- https://subscription.packtpub.com/book/application_development/9781785883378/2/ch02lvl1sec31/exploring-the-ios-filesystem
- https://resources.infosecinstitute.com/ios-application-security-part-10-ios-filesystem-and-forensics/
- https://medium.com/@lucideus/understanding-the-ios-file-system-eee3dc87e455
- https://reincubate.com/support/how-to/mount-iphone-files/
- https://www.youtube.com/watch?v=zjOLDOxZbQU&ab_channel=BillyEllis
- https://www.youtube.com/watch?v=y_PF3d3ZHWM&ab_channel=TroyBradshaw
- https://www.sans.org/blog/checkra1n---part-1---prep/
- https://blog.elcomsoft.com/2020/05/full-file-system-acquisition-for-ios-13-3-1-13-4-and-13-4-1/
- https://www.cellebrite.com/en/blog/ios-breakthrough-enables-lawful-access-for-full-file-system-extraction/

EXPLORING FILESYSTEM IOS

- https://subscription.packtpub.com/book/application_development/9781785883378/2/ch02lvl1sec31/exploring-the-ios-filesystem
- https://resources.infosecinstitute.com/ios-application-security-part-10-ios-filesystem-and-forensics/
- https://medium.com/@lucideus/understanding-the-ios-file-system-eee3dc87e455
- https://reincubate.com/support/how-to/mount-iphone-files/
- https://www.youtube.com/watch?v=zjOLDOxZbQU&ab_channel=BillyEllis
- https://www.youtube.com/watch?v=y_PF3d3ZHWM&ab_channel=TroyBradshaw
- https://www.sans.org/blog/checkra1n---part-1---prep/
- https://blog.elcomsoft.com/2020/05/full-file-system-acquisition-for-ios-13-3-1-13-4-and-13-4-1/
- https://www.cellebrite.com/en/blog/ios-breakthrough-enables-lawful-access-for-full-file-system-extraction/

IOS PINNING

- https://blog.netspi.com/four-ways-to-bypass-ios-ssl-verification-and-certificate-pinning/
- https://stackoverflow.com/questions/58749166/disable-ios-pinning
- https://www.guardsquare.com/en/blog/iOS-SSL-certificate-pinning-bypassing
- http://www.newosxbook.com/articles/CodeSigning.pdf
- https://ios.developreference.com/article/24290405/How+to+bypass+code+signing+in+xcode+4.4+for+iOS+5%3F+%5Bclosed%5D
- https://www.nccgroup.com/us/about-us/newsroom-and-events/blog/2015/january/bypassing-openssl-certificate-pinning-in-ios-apps/

ROOT DETECTION BYPASS

- https://resources.infosecinstitute.com/android-hacking-security-part-8-root-detection-evasion/
- https://resources.infosecinstitute.com/android-root-detection-bypass-reverse-engineering-apk/
- https://medium.com/@sarang6489/root-detection-bypass-by-manual-code-manipulation-5478858f4ad1
- https://www.youtube.com/watch?v=iN0oMBLKxU8&ab_channel=SteveCampbell
- https://julianberton.com/2015/01/30/root-detection-bypass/

LAB MOBILE

- https://medium.com/bugbountywriteup/android-pentesting-lab-4a6fe1a1d2e0
- https://null-byte.wonderhowto.com/how-to/hacking-android-create-lab-for-android-penetration-testing-0186159/
- https://www.theoffensivelabs.com/p/hacking-and-pentesting-android-applications
- https://appsec-labs.com/mobile_pentesting/
- https://github.com/payatu/diva-android
- https://github.com/prateek147/DVIA-v2
- https://github.com/logicalhacking/DVHMA
- https://github.com/OWASP/owasp-mstg/tree/master/Crackmes
- https://github.com/OWASP/igoat
- https://github.com/WaTF-Team/WaTF-Bank

MOBILE BUG BOUNTY

- https://hackerone.com/bug-bounty-programs
- https://security.samsungmobile.com/rewardsProgram.smsb
- https://bugbounty.linecorp.com/en/
- https://www.bugcrowd.com/bug-bounty-list/
- https://www.google.com/about/appsecurity/android-rewards/
- https://developer.apple.com/security-bounty/

CURSOS MOBILE PENTEST – PLATAFORMAS

- https://www.udemy.com/
- https://desecsecurity.com/curso/
- https://www.elearnsecurity.com/
- https://www.eccouncil.org/the-complete-mobile-ethical-hacking-course/
- https://cybrary.it/
- http://blackhat.com/
- https://www.youtube.com/

DORK PARA PESQUISA

mobile pentest filetype:pdf ou pptx

CONCLUSÃO

- Esse é o material de estudo que deixo para vocês, pois o mundo mobile cresce cada vez mais e falta mão de obra para cuidar da segurança desses aplicativos;
- Como dica, eu recomendo que você se aprofunde nas arquiteturas e conceitos básicos dos componentes envolvendo cada um dos sistemas;
- E claro, se você quiser procurar mais conteúdos ou elaborar uma rotina melhor de estudos, consulte ementas de cursos;
- Por ora, meu objetivo foi deixar uma rica fonte de conteúdo para vocês estudarem e se aprofundarem;