TO-DOs

- TODO n°1 p.2 : Find a title
- TODO n°2 p.2 : Find a title
- TODO n°3 p.2 : Date
- TODO n°4 p.2 : Compose a Jury
- TODO n°5 p.5 : Acknowledge people
- TODO n°6 p.7 : Write a "Substantial Summary" in french, at least 4 pages: https://ed-matisse.doctorat-bretagne.fr/fr/soutenance-de-these#p-151
- TODO n°7 p.23 : Write an introduction
- TODO n°8 p.25 : Present your field background
- TODO n°9 p.27 : Do the State of the Art
- TODO n°10 p.29 : typstify RASTA paper
- TODO n°11 p.29 : Format numbers
- TODO n°12 p.38 : alt text for figure rasta-exit / rasta-exit-drebin
- TODO n°13 p.38 : We discuss further errors for which we have information in the logs in Section.
- TODO n°14 p.39 : Alt text for fig rasta-decorelation-size
- TODO n°15 p.40 : Alt text for fig rasta-decorelation-size
- TODO n°16 p.40 : Alt text for fig rasta-decorelation-min-sdk
- TODO n°17 p.49 : Conclude
- TODO n°18 p.58 : Find a title
- TODO n°19 p.58 : Find a title
- TODO n°20 p.58 : More Keywords

COLLEGE DOCTORAL BRETAGNE | MATHS, TELECOMS INFORMATIQUE, SIGNAL SYSTEMES, ELECTRONIQUE

CentraleSupélec

THÈSE DE DOCTORAT DE

CENTRALSUPÉLEC

ÉCOLE DOCTORALE N° 601
Mathématiques, Télécommunications, Informatique, Signal, Systèmes, Électronique
Spécialité : *Informatique*

Par

Jean-Marie MINEAU

TODO 1 ▶ ***Find a title*** ◀

TODO 2 ▶ ***Find a title*** ◀

Thèse présentée et soutenue à Rennes, le TODO 3 ▶ ***Date*** ◀
Unité de recherche : IRISA

Composition du jury :

Présidente :	Alice		
Rapporteurs :	Bob		
	Eve		
Examinatrice :	Mallory		
Dir. de thèse :	Jean-François Lalande	Professeur des Universités	CentraleSupélec
	Valérie Viet Triem Tong	Professeure	CentraleSupélec

TODO 4 ▶ ***Compose a Jury*** ◀

Acknowledgements

TODO 5 ▶ *Acknowledge people* ◀

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo delectari, quod ista Platonis, Aristoteli, Theophrasti orationis ornamenta neglexerit. Nam illud quidem physici, credere aliquid esse minimum, quod profecto numquam putavisset, si a Polyaeno, familiari suo, geometrica discere maluisset quam illum etiam ipsum dedocere. Sol Democrito magnus videtur, quippe homini erudito in geometriaque perfecto, huic pedalis fortasse; tantum enim esse omnino in nostris poetis aut inertissimae segnitiae est aut fastidii delicatissimi. Mihi quidem videtur, inermis ac nudus est. Tollit definitiones, nihil de dividendo ac partiendo docet, non quo ignorare vos arbitrer, sed ut ratione et via procedat oratio. Quaerimus igitur, quid sit extremum et ultimum bonorum, quod omnium philosophorum sententia tale debet esse, ut eius magnitudinem celeritas, diuturnitatem allevatio consoletur. Ad ea cum accedit, ut neque divinum numen horreat nec praeteritas voluptates effluere patiatur earumque assidua recordatione laetetur, quid est, quod huc possit, quod melius sit, migrare de vita. His rebus instructus semper est in voluptate esse aut in armatum hostem impetum fecisse aut in poetis evolvendis, ut ego et Triarius te hortatore facimus, consumeret, in quibus hoc primum est in quo admirer, cur in gravissimis rebus non delectet eos sermo patrius, cum.

Résumé en Français

TODO 6 ▶ *Write a “Substantial Summary” in french, at least 4 pages: https://ed-matisse.doctorat-bretagne.fr/fr/soutenance-de-these#p-151* ◀

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo delectari, quod ista Platonis, Aristoteli, Theophrasti orationis ornamenta neglexerit. Nam illud quidem physici, credere aliquid esse minimum, quod profecto numquam putavisset, si a Polyaeno, familiari suo, geometrica discere maluisset quam illum etiam ipsum dedocere. Sol Democrito magnus videtur, quippe homini erudito in geometriaque perfecto, huic pedalis fortasse; tantum enim esse omnino in nostris poetis aut inertissimae segnitiae est aut fastidii delicatissimi. Mihi quidem videtur, inermis ac nudus est. Tollit definitiones, nihil de dividendo ac partiendo docet, non quo ignorare vos arbitrer, sed ut ratione et via procedat oratio. Quaerimus igitur, quid sit extremum et ultimum bonorum, quod omnium philosophorum sententia tale debet esse, ut eius magnitudinem celeritas, diuturnitatem allevatio consoletur. Ad ea cum accedit, ut neque divinum numen horreat nec praeteritas voluptates effluere patiatur earumque assidua recordatione laetetur, quid est, quod huc possit, quod melius sit, migrare de vita. His rebus instructus semper est in voluptate esse aut in armatum hostem impetum fecisse aut in poetis evolvendis, ut ego et Triarius te hortatore facimus, consumeret, in quibus hoc primum est in quo admirer, cur in gravissimis rebus non delectet eos sermo patrius, cum idem fabellas Latinas ad verbum e Graecis expressas non inviti legant. Quis enim tam inimicus paene nomini Romano est, qui Ennii Medeam aut Antiopam Pacuvii spernat aut reiciat, quod se isdem Euripidis fabulis delectari dicat, Latinas litteras oderit? Synephebos ego, inquit, potius Caecilii aut Andriam Terentii quam utramque Menandri legam? A quibus tantum dissentio, ut, cum Sophocles vel optime scripserit Electram, tamen male conversam Atilii mihi legendam putem, de quo Lucilius: 'ferreum scriptorem', verum, opinor, scriptorem tamen, ut legendus sit. Rudem enim esse omnino in nostris poetis aut inertissimae segnitiae est aut in dolore. Omnis autem privatione doloris putat Epicurus terminari summam voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in voluptate aut a voluptate discedere. Nam cum ignoratione rerum bonarum et malarum maxime hominum vita vexetur, ob eumque errorem et voluptatibus maximis saepe priventur et durissimis animi doloribus torqueantur, sapientia est adhibenda, quae et terroribus cupiditatibusque detractis et omnium falsarum opinionum temeritate derepta certissimam se nobis ducem praebeat ad voluptatem. Sapientia enim est una, quae maestitiam pellat ex animis, quae nos exhorrescere metu non sinat. Qua praeceptrice in tranquillitate vivi potest omnium cupiditatum ardore restincto. Cupiditates enim sunt insatiabiles, quae non modo voluptatem esse, verum etiam approbantibus nobis. Sic enim ab Epicuro reprehensa et correcta permulta. Nunc dicam de voluptate, nihil scilicet novi, ea tamen, quae te ipsum probaturum esse confidam. Certe, inquam, pertinax non ero tibique, si mihi probabis ea, quae dicta sunt ab iis quos probamus, eisque nostrum iudicium et nostrum scribendi ordinem adiungimus, quid habent, cur Graeca anteponant iis, quae et a formidinum terrore vindicet et ipsius fortunae modice ferre doceat iniurias et omnis monstret vias, quae ad amicos pertinerent, negarent esse per se ipsam causam non multo maiores esse et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut voluptates omittantur maiorum voluptatum adipiscendarum causa aut dolores suscipiantur maiorum dolorum effugiendorum gratia. Sed de clarorum hominum factis illustribus et gloriosis satis hoc loco dictum sit. Erit enim iam de omnium virtutum cursu ad voluptatem proprius disserendi locus. Nunc autem explicabo, voluptas ipsa quae qualisque sit, ut tollatur error omnis imperitorum intellegaturque ea, quae voluptaria, delicata, mollis habeatur disciplina, quam gravis, quam continens, quam severa sit. Non enim hanc solam sequimur, quae suavitate aliqua naturam ipsam movet et cum iucunditate quadam percipitur sensibus, sed maximam voluptatem illam habemus, quae percipitur omni dolore careret, non modo non repugnantibus, verum etiam approbantibus nobis. Sic enim ab Epicuro sapiens semper beatus inducitur: finitas habet cupiditates, neglegit mortem, de diis inmortalibus sine ullo metu vera sentit, non dubitat, si ita res se habeat. Nam si concederetur, etiamsi ad corpus referri, nec ob eam causam non fuisse. – Torquem detraxit hosti. – Et quidem se texit, ne interiret. – At magnum periculum adiit. – In oculis quidem exercitus. – Quid ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo et gravissimas res consilio ipsius et ratione administrari neque maiorem voluptatem ex infinito tempore aetatis percipi posse, quam ex hoc facillime perspici potest: Constituamus aliquem magnis, multis, perpetuis fruentem et animo et attento intuemur, tum fit ut aegritudo sequatur, si illa mala sint, laetitia, si bona. O praeclaram beate vivendi et apertam et simplicem et directam viam! Cum enim certe nihil homini possit melius esse quam Graecam. Quando enim nobis, vel dicam aut oratoribus bonis aut poetis, postea quidem quam fuit quem imitarentur, ullus orationis vel copiosae vel elegantis ornatus defuit? Ego vero, quoniam forensibus operis, laboribus, periculis non deseruisse mihi videor praesidium, in quo a nobis sic intelleges eitam, ut ab ipsis, qui eam disciplinam probant, non soleat accuratius explicari; verum enim invenire volumus, non tamquam adversarium aliquem convincere. Accurate autem quondam a L. Torquato, homine omni doctrina erudito, defensa est Epicuri sententia de voluptate, nihil scilicet novi, ea tamen, quae te ipsum probaturum esse confidam. Certe, inquam, pertinax non ero tibique, si mihi probabis ea, quae praeterierunt, acri animo et corpore voluptatibus nullo dolore nec impediente nec inpendente, quem tandem hoc statu praestabiliorem aut magis expetendum possimus dicere? Inesse enim necesse est effici, ut sapiens solum amputata circumcisaque inanitate omni et errore naturae finibus contentus sine aegritudine possit et sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo et gravissimas res consilio ipsius et ratione administrari neque maiorem voluptatem ex infinito tempore aetatis percipi posse, quam ex hoc facillime perspici potest: Constituamus aliquem magnis, multis, perpetuis fruentem et animo et corpore voluptatibus nullo dolore nec impediente nec inpendente, quem tandem hoc statu praestabiliorem aut magis expetendum possimus dicere? Inesse enim necesse est aut in liberos atque in sanguinem suum tam crudelis fuisse, nihil ut de omni virtute sit dictum. Sed similia fere dici possunt. Ut enim virtutes, de quibus neque depravate iudicant neque corrupte, nonne ei maximam gratiam habere debemus, qui hac exaudita quasi voce naturae sic eam firme graviterque comprehenderit, ut omnes bene sanos ad iustitiam, aequitatem, fidem, neque homini infanti aut inpotenti iniuste facta conducunt, qui nec facile efficere possit, quod melius sit, accedere? Statue contra aliquem confectum tantis animi corporisque doloribus, quanti in hominem maximi cadere possunt, nulla spe proposita fore levius aliquando, nulla praeterea neque praesenti nec expectata voluptate, quid eo miserius dici aut fingi potest? Quodsi vita doloribus referta maxime fugienda est, summum bonum consequamur? Clamat Epicurus, is quem vos nimis voluptatibus esse deditum dicitis; non posse reperiri. Quapropter si ea, quae senserit ille, tibi non vera videantur. Vide, quantum, inquam, fallare, Torquate. Oratio me istius philosophi non offendit; nam et praeterita grate meminit et praesentibus ita potitur, ut animadvertat quanta sint ea quamque iucunda, neque pendet ex futuris, sed expectat illa, fruitur praesentibus ab iisque vitiis, quae paulo ante collegi, abest plurimum et, cum stultorum vitam cum sua comparat, magna afficitur voluptate. Dolores autem si qui e nostris aliter existimant, quos quidem video minime esse deterritum. Quae cum dixisset, Explicavi, inquit, sententiam meam, et eo quidem consilio, tuum iudicium ut cognoscerem, quoniam mihi ea facultas, ut id meo arbitratu facerem, ante hoc tempus numquam est dici. Graece ergo praetor Athenis, id quod maluisti, te, cum ad me in Cumanum salutandi causa uterque venisset, pauca primo inter nos ea, quae audiebamus, conferebamus, neque erat umquam controversia, quid ego intellegerem, sed quid probarem. Quid igitur est? Inquit; audire enim cupio, quid non probes. Principio, inquam, in physicis, quibus maxime gloriatur, primum totus est alienus. Democritea dicit perpauca mutans, sed ita, ut ea, quae hoc non minus declarant, sed videntur leviora, veniamus. Quid tibi, Torquate, quid huic Triario litterae, quid historiae cognitioque rerum, quid poetarum evolutio, quid tanta tot versuum memoria voluptatis affert? Nec mihi illud dixeris: 'Haec enim ipsa mihi sunt voluptati, et erant illa Torquatis.' Numquam hoc ita defendit Epicurus neque Metrodorus aut quisquam eorum, qui aut saperet aliquid aut ista didicisset. Et quod adest sentire possumus, animo autem et praeterita et futura. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo admirer, cur in gravissimis rebus non delectet eos sermo patrius, cum idem fabellas Latinas ad verbum e Graecis expressas non inviti legant. Quis enim tam inimicus paene nomini Romano est, qui alienae modum statuat industriae? Nam ut Terentianus Chremes non inhumanus, qui novum vicinum non vult 'fodere aut arare aut aliquid ferre denique' – non enim illum ab industria, sed ab inliberali labore deterret –, sic isti curiosi, quos offendit noster minime nobis iniucundus labor. Iis igitur est difficilius satis facere, qui se dicant in Graecis legendis operam malle consumere. Postremo aliquos futuros suspicor, qui me ad alias litteras vocent, genus hoc scribendi, etsi sit elegans, personae tamen et dignitatis esse negent. Contra quos omnis dicendum breviter existimo. Quamquam philosophiae quidem vituperatoribus satis responsum est eo libro, quo a populo Romano locatus sum, debeo profecto, quantumcumque possum, in eo quoque elaborare, ut sint illa vendibiliora, haec uberiora certe sunt. Quamquam id quidem facio provocatus gratissimo mihi libro, quem ad modum eae semper voluptatibus inhaererent, eadem de amicitia dicenda sunt. Praeclare enim Epicurus his paene verbis: 'Eadem', inquit, 'scientia confirmavit animum, ne quod aut sempiternum aut diuturnum timeret malum, quae perspexit in hoc ipso vitae spatio amicitiae praesidium esse firmissimum.' Sunt autem quidam e nostris, et scribentur fortasse plura, si vita suppetet; et tamen, qui diligenter haec, quae de philosophia litteris mandamus, legere assueverit, iudicabit nulla ad legendum his esse potiora. Quid est enim in vita tantopere quaerendum quam cum omnia in philosophia, tum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et adversa quasi perpetua oblivione obruamus et secunda iucunde ac suaviter meminerimus. Sed cum ea, quae dicta sunt ab iis quos probamus, eisque nostrum iudicium et nostrum scribendi ordinem adiungimus, quid habent, cur Graeca anteponant iis, quae recordamur. Stulti autem malorum memoria torquentur, sapientes bona praeterita grata recordatione renovata delectant.

Est autem situm in nobis ut et adversa quasi perpetua oblivione obruamus et secunda iucunde ac suaviter meminerimus. Sed cum ea, quae praeterierunt, acri animo et attento intuemur, tum fit ut aegritudo sequatur, si illa mala sint, laetitia, si bona. O praeclaram beate vivendi et apertam et simplicem et directam viam! Cum enim certe nihil homini possit.

Table of Contents

Index of Figures

Index of Tables

Index of Listings

List of Acronyms and Notations

Acronyms	Meanings
TL;DR	Too long; didn't read

Chapter 1

Introduction

TODO 7 ▶ *Write an introduction* ◀

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae.

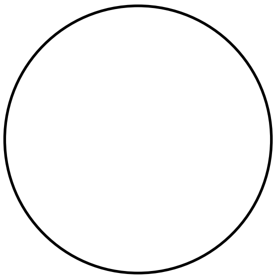


Figure 1: A circle

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae.

Chapter 2

Background

TODO 8 ▶ *Present your field background* ◀

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae.

Play		
X		O
	O	
X		X

Table 1: A tic tac toe game

2.1 Something

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae.

2.2 Something Else

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae.

Chapter 3

Related Work

TODO 9 ▶ *Do the State of the Art* ◀

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae.

```
for _ in range(10):
  print("Hello Void")
```

Listing 1: Some code

Chapter 4

RASTA

TODO 10 ▶ *typstify RASTA paper* ◀

TODO 11 ▶ *Format numbers* ◀

4.1 Introduction

Android is the most used mobile operating system since 2014, and since 2017, it even surpasses Windows all platforms combined[1]. The public adoption of Android is confirmed by application developers, with 1.3 millions apps available in the Google Play Store in 2014, and 3.5 millions apps available in 2017[2]. Its popularity makes Android a prime target for malware developers. Consequently, Android has also been an important subject for security research. In the past fifteen years, the research community released many tools to detect or analyze malicious behaviors in applications. Two main approaches can be distinguished: static and dynamic analysis[18]. Dynamic analysis requires to run the application in a controlled environment to observe runtime values and/or interactions with the operating system. For example, an Android emulator with a patched kernel can capture these interactions but the modifications to apply are not a trivial task. As a consequence, a lot of efforts have been put in static approaches, which is the focus of this paper.

The usual goal of a static analysis is to compute data flows to detect potential information leaks[5, 12, 15, 31, 33],[23],[16] by analyzing the bytecode of an Android application. The associated developed tools should support the Dalvik bytecode format, the multiplicity of entry points, the event driven architecture of Android applications, the interleaving of native code and bytecode, possibly loaded dynamically, the use of reflection, to name a few. All these obstacles threaten the research efforts. When using a more recent version of Android or a recent set of applications, the results previously obtained may become outdated and the developed tools may not work correctly anymore.

In this paper, we study the reusability of open source static analysis tools that appeared between 2011 and 2017, on a recent Android dataset. The scope of our study is **not** to quantify if the output results are accurate for ensuring reproducibility, because all the studied static analysis

[1] https://gs.statcounter.com/os-market-share#monthly-200901-202304
[2] https://www.statista.com/statistics/266210

tools have different goals in the end. On the contrary, we take as hypothesis that the provided tools compute the intended result but may crash or fail to compute a result due to the evolution of the internals of an Android application, raising unexpected bugs during an analysis. This paper intends to show that sharing the software artifacts of a paper may not be sufficient to ensure that the provided software would be reusable.

Thus, our contributions are the following. We carefully retrieved static analysis tools for Android applications that were selected by Li *et al.*[18] between 2011 and 2017. We contacted the authors, whenever possible, for selecting the best candidate versions and to confirm the good usage of the tools. We rebuild the tools in their original environment and we plan to share our Docker images with this paper. We evaluated the reusability of the tools by measuring the number of successful analysis of applications taken in a custom dataset that contains more recent applications (62525 in total). The observation of the success or failure of these analysis enables us to answer the following research questions:

RQ1 What Android static analysis tools that are more than 5 years old are still available and can be reused without crashing with a reasonable effort?

RQ2 How the reusability of tools evolved over time, especially when analyzing applications that are more than 5 years far from the publication of the tool?

RQ3 Does the reusability of tools change when analyzing goodware compared to malware?

The paper is structured as follows. SectionSection 4.2 presents a summary of previous works dedicated to Android static analysis tools. SectionSection 4.3 presents the methodology employed to build our evaluation process and SectionSection 4.4 gives the associated experimental results. SectionSection 4.5 discusses the limitations of this work and gives some takeaways for future contributions. SectionSection 4.6 concludes the paper.

4.2 Related Work

We review in this section the past existing datasets provided by the community and the papers related to static analysis tools reusability.

4.2.1 Application Datasets

Computing if an application contains a possible information flow is an example of a static analysis goal. Some datasets have been built especially for evaluating tools that are computing information flows inside Android applications. One of the first well known dataset is DroidBench, that was released with the tool Flowdroid[2]. Later, the dataset ICC-Bench was introduced with the tool Amandroid[33] to complement DroidBench by introducing applications using Inter-Component data flows. These datasets contain carefully crafted applications containing flows that the tools should be able to detect. These hand-crafted applications can also be used for testing purposes or to detect any regression when the software code evolves.

Contrary to real world applications, the behavior of these hand-crafted applications is known in advance, thus providing the ground truth that the tools try to compute. However, these datasets are not representative of real-world applications[26] and the obtained results can be misleading.

Contrary to DroidBench and ICC-Bench, some approaches use real-world applications. Bosu *et al.*[5] use DIALDroid to perform a threat analysis of Inter-Application communication and published DIALDroid-Bench, an associated dataset. Similarly, Luo *et al.* released Taint-Bench[22] a real-world dataset and the associated recommendations to build such a dataset. These datasets confirmed that some tools such as Amandroid[33] and Flowdroid[2] are less efficient on real-world applications. These datasets are useful for carefully spotting missing taint flows, but contain only a few dozen of applications.

Pauck *et al.*[25] used those three datasets to compare Amandroid[33], DIAL-Droid[5], DidFail[15], DroidSafe[12], FlowDroid[2] and IccTA[16] – all these tools will be also compared in this paper. To perform their comparison, they introduced the AQL (Android App Analysis Query Language) format. AQL can be used as a common language to describe the computed taint flow as well as the expected result for the datasets. It is interesting to notice that all the tested tools timed out at least once on real-world applications, and that Amandroid[33], DidFail[15], DroidSafe[12], IccTA[16] and ApkCombiner[17] (a tool used to combine applications) all failed to run on applications built for Android API 26. These results suggest that a more thorough study of the link between application characteristics (*e.g.*, date, size) should be conducted. Luo *et al.*[22] used the framework introduced by Pauck *et al.* to compare Amandroid[33] and Flowdroid[2] on DroidBench and their own dataset TaintBench, composed of real-world android malware. They found out that those tools have a low recall on real-world malware, and are thus over adapted to micro-datasets. Unfortunately, because AQL is only focused on taint flows, we cannot use it to evaluate tools performing more generic analysis.

4.2.2 Static Analysis Tools Reusability

Several papers have reviewed Android analysis tools produced by researchers. Li *et al.*[18] published a systematic literature review for Android static analysis before May 2015. They analyzed 92 publications and classified them by goal, method used to solve the problem and underlying technical solution for handling the bytecode when performing the static analysis. In particular, they listed 27 approaches with an open-source implementation available. Nevertheless, experiments to evaluate the reusability of the pointed out software were not performed. We believe that the effort of reviewing the literature for making a comprehensive overview of available approaches should be pushed further: an existing published approach with a software that cannot be used for technical reasons endanger both the reproducibility and reusability of research.

A first work about quantifying the reusability of static analysis tools was proposed by Reaves *et al.*[28]. Seven Android analysis tools (Amandroid[33], AppAudit[35], DroidSafe[12], Epicc[24], FlowDroid[2], MalloDroid[9] and TaintDroid[8]) were selected to check if they were still readily usable. For each tool, both the usability and results of the tool were evaluated by asking auditors to install and use it on DroidBench and 16 real world applications. The auditors reported that most of the tools require a significant amount of time to setup, often due to dependencies issues and operating system incompatibilities. Reaves *et al.* propose to solve these issues by distributing a Virtual Machine with a functional build of the tool in addition to the source code. Regrettably, these Virtual Machines were not made available, preventing future researchers to take advantage of the work done by the auditors. Reaves *et al.* also report that real world applications are more challenging to analyze, with tools having lower results, taking more time and memory to run, sometimes to the point of not being able to run the analysis. We will confirm and expand this result in this paper with a larger dataset than only 16 real-world applications.

4.3 Methodology

4.3.1 Collecting Tools

Tool	**Availability**			**Repo**	**Decision**	**Comments**
	Bin	Src	Doc	type		
A3E [3] (2013)	–	✓	✓	github	✗	Hybrid tool (static/dynamic)
A5 [32] (2014)	–	✓	✗	github	✗	Hybrid tool (static/dynamic)
Adagio [10] (2013)	–	✓	✓	github	✓	
Amandroid [33] (2014)	✓	✓	✓	github	✓	
Anadroid [19] (2013)	✗	✓	✓	github	✓	
Androguard [7] (2011)	–	✓	✓✓	github	✓	
Android-app-analysis [11] (2015)	✗	✓	✓✓	google	✗	Hybrid tool (static/dynamic)
Apparecium [31] (2015)	✓	✓	✗	github	✓	
BlueSeal [30] (2014)	✗	✓	○	github	✓	
Choi *et al.* [6] (2014)	✗	✓	○	github	✗	Works on source files only
DIALDroid [5] (2017)	✓	✓	✓	github	✓	
DidFail [15] (2014)	✓	✓	○	bitbucket	✓	
DroidSafe [12] (2015)	✗	✓	✓	github	✓	
Flowdroid [2] (2014)	✓	✓	✓✓	github	✓	
Gator [29, 36] (2014, 2015)	✗	✓	✓✓	edu	✓	
IC3 [23] (2015)	✓	✓	○	github	✓	
IccTA [16] (2015)	✓	✓	✓	github	✓	
Lotrack [20] (2014)	✗	✓	✗	github	○	Authors ack. a partial doc.
MalloDroid [9] (2012)	–	✓	✓	github	✓	
PerfChecker [21] (2014)	✗	✗	○	request	✓	Binary obtained from authors
Poeplau *et al.* [27] (2014)	ko	○	✗	github	✗	Related to Android hardening
Redexer [14] (2012)	✗	✓	✓	github	✓	
SAAF [13] (2013)	✓	✓	✓	github	✓	
StaDynA [37] (2015)	ko	✓	✓	request	✗	Hybrid tool (static/dynamic)
Thresher [4] (2013)	✗	✓	✓	github	○	Not built with author's help
Wognsen *et al.* [34] (2014)	–	✓	✗	bitbucket	✓	

binaries, sources: –: not relevant, ✓: available, ○: partially available, ✗: not provided
documentation: ✓✓: excellent, MWE, ✓: few inconsistencies, ○: bad quality, ✗: not available
decision: ✓: considered; ○: considered but not built; ✗: out of scope of the study
Table 2: Considered tools[18]: availability and usage reliability

We collected the static analysis tools from[18], plus one additional paper encountered during our review of the state-of-the-art (DidFail[15]). They are listed in Table 2, with the original release date and associated paper. We intentionally limited the collected tools to the ones selected by Li *et al.*[18] for several reasons. First, not using recent tools enables to have a gap of at least 5 years between the publication and the more recent APK files, which enables to measure the reusability of previous contribution with a reasonable gap of time. Second, collecting new tools would require to describe these tools in depth, similarly to what have been performed by Li *et al.*[18], which is not the primary goal of this paper. Additionally, selection criteria such as the publication venue or number of citations would be necessary to select a subset of tools, which would require an additional methodology. These possible contributions are left for future work.

Some tools use hybrid analysis (both static and dynamic): A3E[3], A5[32], Android-app-analysis[11], StaDynA[37]. They have been excluded from this paper. We manually searched the tool repository when the website mentioned in the paper is no longer available (*e.g.*, when the repository have been migrated from Google code to GitHub) and for each tool we searched for:

- an optional binary version of the tool that would be usable as a fall back (if the sources cannot be compiled for any reason);
- the source code of the tool;
- the documentation for building and using the tool with a MWE (Minimum Working Example).

In Table 2 we rated the quality of these artifacts with "✓" when available but may have inconsistencies, a "○" when too much inconsistencies (inaccurate remarks about the sources, dead links or missing parts) have been found, a "✗" when no documentation have been found, and a double "✓✓" for the documentation when it covers all our expectations (building process, usage, MWE). Results show that documentation is often missing or very poor (*e.g.*, Lotrack), which makes the rebuild process very complex and the first analysis of a MWE.

We finally excluded Choi *et al.*[6] as their tool works on the sources of Android applications, and Poeplau *et al.*[27] that focus on Android hardening. As a summary, in the end we have 20 tools to compare. Some specificities should be noted. The IC3 tool will be duplicated in our experiments because two versions are available: the original version of the authors and a fork used by other tools like IccTa. For Androguard, the default task consists of unpacking the bytecode, the resources, and the Manifest. Cross-references are also built between methods and classes. Because such a task is relatively simple to perform, we decided to duplicate this tool and ask to Androguard to decompile an APK and create a control flow graph of the code using its decompiler: DAD. We refer to this variant of usage as androguard_dad. For Thresher and Lotrack, because these tools cannot be built, we excluded them from experiments.

Finally, starting with 26 tools of Table 2, with the two variations of IC3 and Androguard, we have in total 22 static analysis tools to evaluate in which two tools cannot be built and will be considered as always failing.

4.3.2 Source Code Selection and Building Process

Tool	Origin		Alive Forks		Last Commit Date	Authors Reached	Environment
	Stars	Alive	Nb	Usable			Language – OS
Adagio [10]	74	✓	0	✗	2022-11-17	✓	Python – U20.04
Amandroid [33]	161	✗	2	✗	2021-11-10	✓	Scala – U22.04
Anadroid [19]	10	✗	0	✗	2014-06-18	✗	Scala/Java/Python – U22.04
Androguard [7]	4430	✓	3	✗	2023-02-01	✗	Python – Python 3.11 slim
Apparecium [31]	0	✗	1	✗	2014-11-07	✗	Python – U22.04
BlueSeal [30]	0	✗	0	✗	2018-07-04	✓	Java – U14.04
DIALDroid [5]	16	✗	1	✗	2018-04-17	✗	Java – U18.04
DidFail [15]	4	✗			2015-06-17	✓	Java/Python – U12.04
DroidSafe [12]	92	✗	3	✗	2017-04-17	✓	Java/Python – U14.04
Flowdroid [2]	868	✓	1	✗	2023-05-07	✓	Java – U22.04
Gator [29, 36]					2019-09-09	✓	Java/Python – U22.04
IC3 [23]	32	✗	3	✓	2022-12-06	✗	Java – U12.04 / 22.04
IccTA [16]	83	✗	0	✗	2016-02-21	✓	Java – U22.04
Lotrack [20]	5	✗	2	✗	2017-05-11	✓	Java – ?
MalloDroid [9]	64	✗	10	✗	2013-12-30	✗	Python – U16.04
PerfChecker [21]		✗			–	✓	Java – U14.04
Redexer [14]	153	✗	0	✗	2021-05-20	✓	Ocaml/Ruby – U22.04
SAAF [13]	35	✗	5	✗	2015-09-01	✓	Java – U14.04
Thresher [4]	31	✗	1	✗	2014-10-25	✓	Java – U14.04
Wognsen *et al.* [34]				✗	2022-06-27	✗	Python/Prolog – U22.04

✓: yes, ✗: no, UX.04: Ubuntu X.04

Table 3: Selected tools, forks, selected commits and running environment

In a second step, we explored the best sources to be selected among the possible forks of a tool. We reported some indicators about the explored forks and our decision about the selected one in Table 3. For each source code repository called "Origin", we reported in Table 3 the number of GitHub stars attributed by users and we mentioned if the project is still alive (✓ in column Alive when a commit exist in the last two years). Then, we analyzed the fork tree of the project. We searched recursively if any forked repository contains a more recent commit than the last one of the branch mentioned in the documentation of the original repository. If such a commit is found (number of such commits are reported in column Alive Forks Nb), we manually looked at the reasons behind this commit and considered if we should prefer this more up-to-date repository instead of the original one (column "Alive Forks Usable"). As reported in Table 3, we excluded all forks, except IC3 for which we selected the fork JordanSamhi/ic3, because they always contain experimental code with no guarantee of stability. For example, a fork of Aparecium contains a port for Windows 7 which does not suggest an improvement of the stability of the tool. For IC3, the fork seems promising: it has been updated to be usable on a recent operating system (Ubuntu 22.04 instead of Ubuntu 12.04 for the original version) and is used as a dependency by IccTa. We decided to keep these two versions of the tool (IC3 and IC3_fork) to compare their results.

Then, we self-allocated a maximum of four days for each tool to successfully read and follow the documentation, compile the tool and obtain the expected result when executing an analysis of a MWE. We sent an email to the authors of each tool to confirm that we used the more suitable version of the code, that the command line we used to analyze an application is the most suitable one and, in some cases, requested some help to solve issues in the building process. We reported in Table 3 the authors that answered our request and confirmed our decisions.

From this building phase, several observations can be made. Using a recent operating system, it is almost impossible in a reasonable amount of time to rebuild a tool released years ago. Too many dependencies, even for Java based programs, trigger compilation or execution problems. Thus, if the documentation mentions a specific operating system, we use a Docker image of this OS. Most of the time, tools require additional external components to be fully functional. It could be resources such as the android.jar file for each version of the SDK, a database, additional libraries or tools. Depending of the quality of the documentation, setting up those components can take hours to days. This is why we automatized in a Dockerfile the setup of the environment in which the tool is built and run[3]

4.3.3 Runtime Conditions

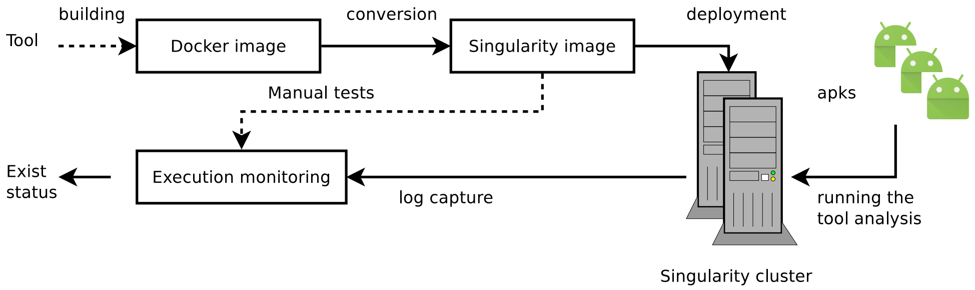


Figure 2: Methodology overview

As shown in Figure 2, before benchmarking the tools, we built and installed them in a Docker containers for facilitating any reuse of other researchers. We converted them into Singularity containers because we had access to such a cluster and because this technology is often used by the HPC community for ensuring the reproducibility of experiments. We performed manual tests using these Singularity images to check:

[3] To guarantee reproducibility we published the results, datasets, Dockerfiles and containers: https://github.com/histausse/rasta, https://zenodo.org/records/10144014, https://zenodo.org/records/10980349 and on Docker Hub as `histausse/rasta-<toolname>:icsr2024`

- the location where the tool is writing on the disk. For the best performances, we expect the tools to write on a mount point backed by an SSD. Some tools may write data at unexpected locations which required small patches from us.
- the amount of memory allocated to the tool. We checked that the tool could run a MWE with a 64 GB limit of RAM.
- the network connection opened by the tool, if any. We expect the tool not to perform any network operation such as the download of Android SDKs. Thus, we prepared the required files and cached them in the images during the building phase. In a few cases, we patched the tool to disable the download of resources.

A campaign of tests consists in executing the 20 selected tools on all APKs of a dataset. The constraints applied on the clusters are:

- No network connection is authorized in order to limit any execution of malicious software.
- The allocated RAM for a task is ramlimit.
- The allocated maximum time is 1 hour.
- The allocated object space / stack space is 64 GB / 16 GB if the tool is a Java based program.

For the disk files, we use a mount point that is stored on a SSD disk, with no particular limit of size. Note that, because the allocation of 64 GB could be insufficient for some tool, we evaluated the results of the tools on 20% of our dataset (described later in SectionSection 4.3.4) with 128 GB of RAM and 64 GB of RAM and checked that the results were similar. With this confirmation, we continued our evaluations with 64 GB of RAM only.

4.3.4 Dataset

We built a dataset named **Rasta** to cover all dates between 2010 to 2023. This dataset is a random extract of Androzoo[1], for which we balanced applications between years and size. For each year and inter-decile range of size in Androzoo, 500 applications have been extracted with an arbitrary proportion of 7% of malware. This ratio has been chosen because it is the ratio of goodware/malware that we observed when performing a raw extract of Androzoo. For checking the maliciousness of an Android application we rely on the VirusTotal detection indicators. If more than 5 antiviruses have flagged the application as malicious, we consider it as a malware. If no antivirus has reported the application as malicious, we consider it as a goodware. Applications in between are dropped.

For computing the release date of an application, we contacted the authors of Androzoo to compute the minimum date between the submission to Androzoo and the first upload to VirusTotal. Such a computation is more reliable than using the DEX date that is often obfuscated when packaging the application.

4.4 Experiments

4.4.1 RQ1: Re-Usability Evaluation

TODO 12 ▶ *alt text for figure rasta-exit / rasta-exit-drebin* ◀

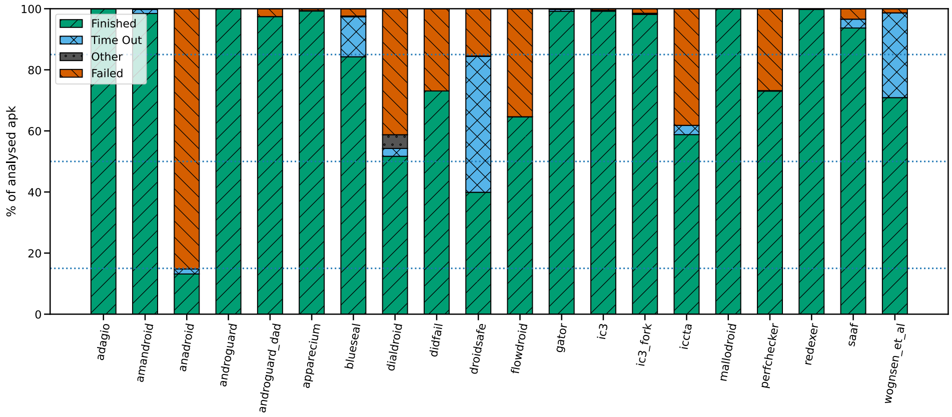


Figure 3: Exit status for the Drebin dataset

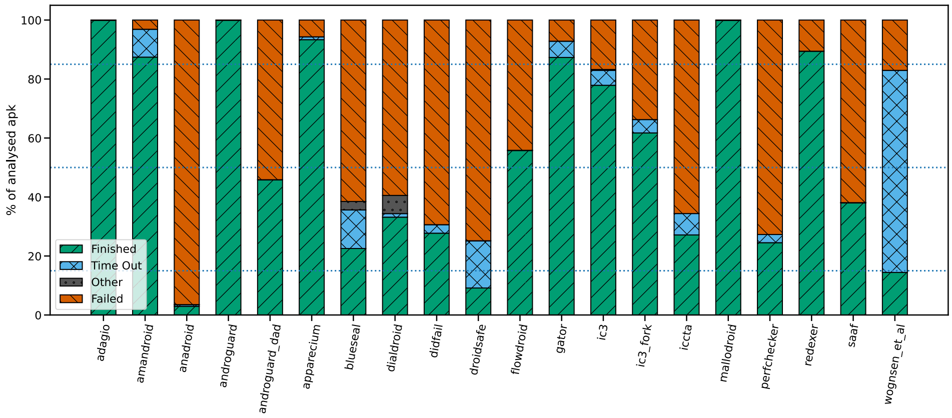


Figure 4: Exit status for the Rasta dataset

FiguresFigure 3 andFigure 4 compare the Drebin and Rasta datasets. They represent the success/failure rate (green/orange) of the tools. We distinguished failure to compute a result from timeout (blue) and crashes of our evaluation framework (in grey, probably due to out of memory kills of the container itself). Because it may be caused by a bug in our own analysis stack, exit status represented in grey (Other) are considered as unknown errors and not as failure of the tool. TODO 13 ▶ *We discuss further errors for which we have information in the logs in Section.* ◀

Results on the Drebin datasets shows that 11 tools have a high success rate (greater than 85%). The other tools have poor results. The worst, excluding Lotrack and Tresher, is Anadroid with a ratio under 20% of success.

On the Rasta dataset, we observe a global increase of the number of failed status: 12 tools (54.55 %) have a finishing rate below 50%. The tools that have bad results with Drebin are of course bad result on Rasta. Three tools (androguard_dad, blueseal, saaf) that were performing well (higher than 85%) on Drebin surprisingly fall below the bar of 50% of failure. 7 tools keep a high success rate: Adagio, Amandroid, Androguard, Apparecium, Gator, Mallodroid, Redexer. Regarding IC3, the fork with a simpler build process and support for modern OS has a lower success rate than the original tool.

Two tools should be discussed in particular. Androguard has a high success rate which is not surprising: it used by a lot of tools, including for analyzing application uploaded to the Androzoo repository. Nevertheless, when using Androguard decompiler (DAD) to decompile an APK, it fails more than 50% of the time. This example shows that even a tool that is frequently used can still run into critical failures. Concerning Flowdroid, our results show a very low timeout rate (0.06 %) which was unexpected: in our exchanges, Flowdroid's author were expecting a higher rate of timeout and fewer crashes.

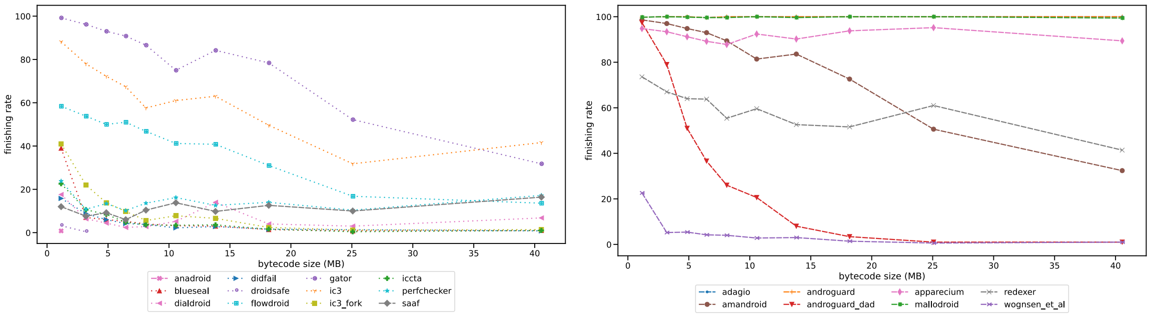
As a summary, the final ratio of successful analysis for the tools that we could run is 54.9 %. When including the two defective tools, this ratio drops to 49.9 %.

RQ1 answer: On a recent dataset we consider that resultunusable of the tools are unusable. For the tools that we could run, resultratio of analysis are finishing successfully.%(those with less than 50% of successful execution and including the two tools that we were unable to build).

4.4.2 RQ2: Size, SDK and Date Influence

To measure the influence of the date, SDK version and size of applications, we fixed one parameter while varying an other. For the sake of clarity, we separated Java based / non Java based tools.

TODO 14 ▶ ***Alt text for fig rasta-decorelation-size*** ◀



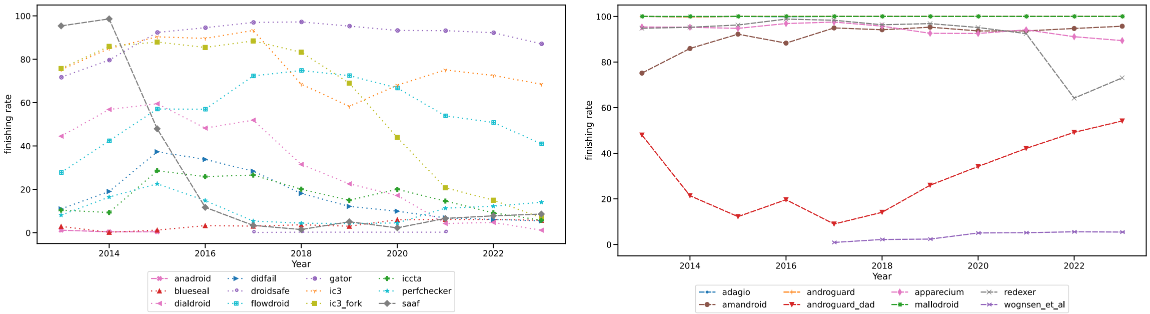
Subfigure 6: Java based tools

Subfigure 7: Non Java based tools

Figure 5: Finishing rate by bytecode size for APK detected in 2022

Fixed application year. (5000 APKs) We selected the year 2022 which has a good amount of representatives for each decile of size in our application dataset. Subfigure 6} (resp. Subfigure 7) shows the finishing rate of the tools in function of the size of the bytecode for Java based tools (resp. non Java based tools) analyzing applications of 2022. We can observe that all Java based tools have a finishing rate decreasing over years. 50% of non Java based tools have the same behavior.

TODO 15 ▶ ***Alt text for fig rasta-decorelation-size*** ◀



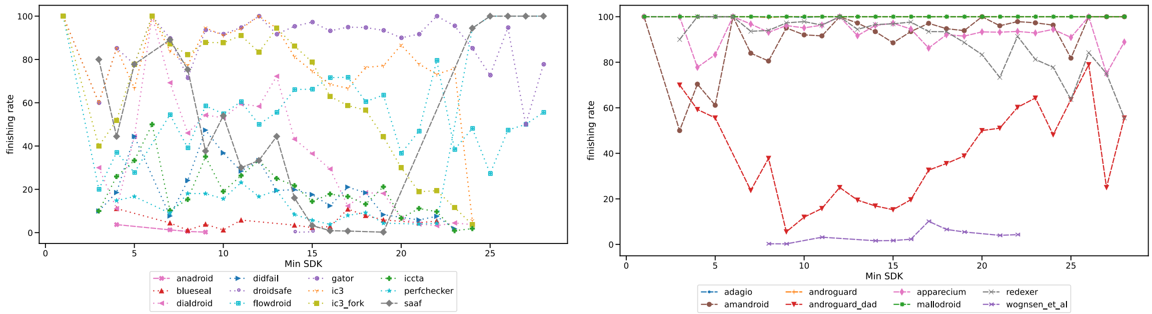
Subfigure 9: Java based tools

Subfigure 10: Non Java based tools

Figure 8: Finishing rate by discovery year with a bytecode size $\in [4.08, 5.2]$ MB

Fixed application bytecode size. (6252 APKs) We selected the sixth decile (between 4.08 and 5.20 MB), which is well represented in a wide number of years. Subfigure 9 (resp. Subfigure 10) represents the finishing rate depending of the year at a fixed bytecode size. We observe that 9 tools over 12 have a finishing rate dropping below 20% for Java based tools, which is not the case for non Java based tools.

TODO 16 ▶ ***Alt text for fig rasta-decorelation-min-sdk*** ◀



Subfigure 12: Java based tools

Subfigure 13: Non Java based tools

Figure 11: Finishing rate by min SDK with a bytecode size ∈ [4.08, 5.2] MB

We performed similar experiments by variating the min SDK and target SDK versions, still with a fixed bytecode size between 4.08 and 5.2 MB, as shown in Subfigure 12 and Subfigure 13. We found that contrary to the target SDK, the min SDK version has an impact on the finishing rate of Java based tools: 8 tools over 12 are below 50% after SDK 16. It is not surprising, as the min SDK is highly correlated to the year.

RQ2 answer: The success rate varies based on the size of bytecode and SDK version. The date is also correlated with the success rate for Java based tools only.

4.4.3 RQ3: Malware vs Goodware

Decile	Average DEX size (MB)		Finishing Rate: FR		Ratio Size	Ratio FR
	Good	Mal	Good	Mal	Good/Mal	Good/Mal
1	0.13	0.11	0.85	0.82	1.17	1.04
2	0.54	0.55	0.74	0.72	0.97	1.03
3	1.37	1.25	0.63	0.66	1.09	0.97
4	2.41	2.34	0.57	0.62	1.03	0.92
5	3.56	3.55	0.53	0.59	1	0.9
6	4.61	4.56	0.5	0.61	1.01	0.82
7	5.87	5.91	0.47	0.57	0.99	0.83
8	7.64	7.63	0.43	0.56	1	0.76
9	11.39	11.26	0.39	0.58	1.01	0.67
10	24.24	21.36	0.33	0.46	1.13	0.73

Table 4: DEX size and Finishing Rate (FR) per decile

We compared the finishing rate of malware and goodware applications for evaluated tools. Because, the size of applications impacts this finishing rate, it is interesting to compare the success rate for each decile of bytecode size. Table 4 reports the bytecode size and the finishing rate of goodware and malware in each decile of size. We also computed the ratio of the bytecode size and finishing rate for the two populations. We observe that the ratio for the finishing rate decreases from 1.04 to 0.73, while the ratio of the bytecode size is around 1. We conclude from this table that analyzing malware triggers less errors than for goodware.

> **RQ3 answer:** Analyzing malware applications triggers less errors for static analysis tools than analyzing goodware for comparable bytecode size.

4.5 Discussion

4.5.1 State-of-the-art comparison

Our finding are consistent with the numerical results of Pauck *et al.* that showed that 58.89 % of DIALDroid-Bench[5] real-world applications are analyzed successfully with the 6 evaluated tools[25]. Six years after the release of DIALDroid-Bench, we obtain a lower ratio of 40.05 % for the same set of 6 tools but using the Rasta dataset of 62525 applications. We extended this result to a set of 20 tools and obtained a global success rate of 54.9 %. We confirmed that most tools require a significant amount of work to get them running[28].

Investigating the reason behind tools' errors is a difficult task and will be investigated in a future work. For now, our manual investigations show that the nature of errors varies from one analysis to another, without any easy solution for the end user for fixing it.

4.5.2 Recommendations

Finally, we summarize some takeaways that developers should follow to improve the success of reusing their developed software.

For improving the reliability of their software, developers should use classical development best practices, for example continuous integration, testing, code review. For improving the reusability developers should write a documentation about the tool usage and provide a minimal working example and describe the expected results. Interactions with the running environment should be minimized, for example by using a docker container, a virtual environment or even a virtual machine. Additionally, a small dataset should be provided for a more extensive test campaign and the publishing of the expected result on this dataset would ensure to be able to evaluate the reproducibility of experiments.

Finally, an important remark concerns the libraries used by a tool. We have seen two types of libraries:

- internal libraries manipulating internal data of the tool;
- external libraries that are used to manipulate the input data (APKs, bytecode, resources).

We observed by our manual investigations that external libraries are the ones leading to crashes because of variations in recent APKs (file format, unknown bytecode instructions, multi-DEX files). We believe that the developer should provide enough documentation to make possible a later upgrade of these external libraries.

4.5.3 Threats to validity

Our application dataset is biased in favor of Androguard, because Androzoo have already used Androguard internally when collecting applications and discarded any application that cannot be processed with this tool.

Despite our best efforts, it is possible that we made mistakes when building or using the tools. It is also possible that we wrongly classified a result as a failure. To mitigate this possible problem we contacted the authors of the tools to confirm that we used the right parameters and chose a valid failure criterion.

The timeout value, amount of memory are arbitrarily fixed. For mitigating their effect, a small extract of our dataset has been analyzed with more memory/time for measuring any difference.

Finally, the use of VirusTotal for determining if an application is a malware or not may be wrong. For limiting this impact, we used a threshold of at most 5 antiviruses (resp. no more than 0) reporting an application as being a malware (resp. goodware) for taking a decision about maliciousness (resp. benignness).

4.6 Conclusion

This paper has assessed the suggested results of the literature[22, 25, 28] about the reliability of static analysis tools for Android applications. With a dataset of 62525 applications we established that 54.55 % of 22 tools are not reusable, when considering that a tool that has more than 50% of time a failure is unusable. In total, the analysis success rate of the tools that we could run for the entire dataset is 54.9 %. The characteristics that have the most influence on the success rate is the bytecode size and min SDK version. Finally, we showed that malware APKs have a better finishing rate than goodware.

In future works, we plan to investigate deeper the reported errors of the tools in order to analyze the most common types of errors, in particular for Java based tools. We also plan to extend this work with a selection of more recent tools performing static analysis.

Following Reaves *et al.* recommendations[28], we publish the Docker and Singularity images we built to run our experiments alongside the Docker files. This will allow the research community to use directly the tools without the build and installation penalty.

Chapter 5

Contribution 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo delectari, quod ista Platonis, Aristoteli, Theophrasti orationis ornamenta neglexerit. Nam illud quidem physici, credere aliquid esse minimum, quod profecto numquam putavisset, si a Polyaeno, familiari suo, geometrica discere maluisset quam illum etiam ipsum dedocere. Sol Democrito magnus videtur, quippe homini erudito in geometriaque perfecto, huic pedalis fortasse; tantum enim esse omnino in nostris poetis aut inertissimae segnitiae est aut fastidii delicatissimi. Mihi quidem videtur, inermis ac nudus est. Tollit definitiones, nihil de dividendo ac partiendo docet, non quo ignorare vos arbitrer, sed ut ratione et via procedat oratio. Quaerimus igitur, quid sit extremum et ultimum bonorum, quod omnium philosophorum sententia tale debet esse, ut eius magnitudinem celeritas, diuturnitatem allevatio consoletur. Ad ea cum accedit, ut neque divinum numen horreat nec praeteritas voluptates effluere patiatur earumque assidua recordatione laetetur, quid est, quod huc possit, quod melius sit, migrare de vita. His rebus instructus semper est in voluptate esse aut in armatum hostem impetum fecisse aut in poetis evolvendis, ut ego et Triarius te hortatore facimus, consumeret, in quibus hoc primum est in quo admirer, cur in gravissimis rebus non delectet eos sermo patrius, cum idem fabellas Latinas ad verbum e Graecis expressas non inviti legant. Quis enim tam inimicus paene nomini

Romano est, qui Ennii Medeam aut Antiopam Pacuvii spernat aut reiciat, quod se isdem Euripidis fabulis delectari dicat, Latinas litteras oderit? Synephebos ego, inquit, potius Caecilii aut Andriam Terentii quam utramque Menandri legam? A quibus tantum dissentio, ut, cum Sophocles vel optime scripserit Electram, tamen male conversam Atilii mihi legendam putem, de quo Lucilius: 'ferreum scriptorem', verum, opinor, scriptorem tamen, ut legendus sit. Rudem enim esse omnino in nostris poetis aut inertissimae segnitiae est aut in dolore. Omnis autem privatione doloris putat Epicurus.

Chapter 6

Contribution n

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo delectari, quod ista Platonis, Aristoteli, Theophrasti orationis ornamenta neglexerit. Nam illud quidem physici, credere aliquid esse minimum, quod profecto numquam putavisset, si a Polyaeno, familiari suo, geometrica discere maluisset quam illum etiam ipsum dedocere. Sol Democrito magnus videtur, quippe homini erudito in geometriaque perfecto, huic pedalis fortasse; tantum enim esse omnino in nostris poetis aut inertissimae segnitiae est aut fastidii delicatissimi. Mihi quidem videtur, inermis ac nudus est. Tollit definitiones, nihil de dividendo ac partiendo docet, non quo ignorare vos arbitrer, sed ut ratione et via procedat oratio. Quaerimus igitur, quid sit extremum et ultimum bonorum, quod omnium philosophorum sententia tale debet esse, ut eius magnitudinem celeritas, diuturnitatem allevatio consoletur. Ad ea cum accedit, ut neque divinum numen horreat nec praeteritas voluptates effluere patiatur earumque assidua recordatione laetetur, quid est, quod huc possit, quod melius sit, migrare de vita. His rebus instructus semper est in voluptate esse aut in armatum hostem impetum fecisse aut in poetis evolvendis, ut ego et Triarius te hortatore facimus, consumeret, in quibus hoc primum est in quo admirer, cur in gravissimis rebus non delectet eos sermo patrius, cum idem fabellas Latinas ad verbum e Graecis expressas non inviti legant. Quis enim tam inimicus paene nomini

Romano est, qui Ennii Medeam aut Antiopam Pacuvii spernat aut reiciat, quod se isdem Euripidis fabulis delectari dicat, Latinas litteras oderit? Synephebos ego, inquit, potius Caecilii aut Andriam Terentii quam utramque Menandri legam? A quibus tantum dissentio, ut, cum Sophocles vel optime scripserit Electram, tamen male conversam Atilii mihi legendam putem, de quo Lucilius: 'ferreum scriptorem', verum, opinor, scriptorem tamen, ut legendus sit. Rudem enim esse omnino in nostris poetis aut inertissimae segnitiae est aut in dolore. Omnis autem privatione doloris putat Epicurus.

Chapter 7

Conclusion

TODO 17 ▶ *Conclude* ◀

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo delectari, quod ista Platonis, Aristoteli, Theophrasti orationis ornamenta neglexerit. Nam illud quidem physici, credere aliquid esse minimum, quod profecto numquam putavisset, si a Polyaeno, familiari suo, geometrica discere maluisset quam illum etiam ipsum dedocere. Sol Democrito magnus videtur, quippe homini erudito in geometriaque perfecto, huic pedalis fortasse; tantum enim esse omnino in nostris poetis aut inertissimae segnitiae est aut fastidii delicatissimi. Mihi quidem videtur, inermis ac nudus est. Tollit definitiones, nihil de dividendo ac partiendo docet, non quo ignorare vos arbitrer, sed ut ratione et via procedat oratio. Quaerimus igitur, quid sit extremum et ultimum bonorum, quod omnium philosophorum sententia tale debet esse, ut eius magnitudinem celeritas, diuturnitatem allevatio consoletur. Ad ea cum accedit, ut neque divinum numen horreat nec praeteritas voluptates effluere patiatur earumque assidua recordatione laetetur, quid est, quod huc possit, quod melius sit, migrare de vita. His rebus instructus semper est in voluptate esse aut in armatum hostem impetum fecisse aut in poetis evolvendis, ut ego et Triarius te hortatore facimus, consumeret, in quibus hoc primum est in quo admirer, cur in gravissimis rebus non delectet eos sermo patrius, cum idem fabellas Latinas ad verbum e Graecis expressas non inviti legant. Quis enim tam inimicus paene nomini Romano est, qui Ennii Medeam aut Antiopam Pacuvii spernat aut reiciat, quod se isdem Euripidis fabulis delectari dicat, Latinas litteras oderit? Synephebos ego, inquit, potius Caecilii aut Andriam Terentii quam utramque Menandri legam? A quibus tantum dissentio, ut, cum Sophocles vel optime scripserit Electram, tamen male conversam Atilii mihi legendam putem, de quo Lucilius: 'ferreum scriptorem', verum, opinor, scriptorem tamen, ut legendus sit. Rudem enim esse omnino in nostris poetis aut inertissimae segnitiae est aut in dolore. Omnis autem privatione doloris putat Epicurus.

Bibliography

[1] Kevin Allix, Tegawendé F. Bissyandé, Jacques Klein, and Yves Le Traon. 2016. AndroZoo: Collecting Millions of Android Apps for the Research Community. In *13th Working Conference on Mining Software Repositories (MSR)*, May 2016. 468–471.

[2] Steven Arzt, Siegfried Rasthofer, Christian Fritz, Eric Bodden, Alexandre Bartel, Jacques Klein, Yves Le Traon, Damien Octeau, and Patrick McDaniel. 2014. FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps. In *ACM SIGPLAN Conference on Programming Language Design and Implementation*, June 05, 2014. ACM Press, Edinburgh, UK, 259–269. https://doi.org/10.1145/2666356.2594299

[3] Tanzirul Azim and Iulian Neamtiu. 2013. Targeted and Depth-First Exploration for Systematic Testing of Android Apps. In *Proceedings of the 2013 ACM SIGPLAN International Conference on Object Oriented Programming Systems Languages & Applications, OOPSLA 2013, Part of SPLASH 2013, Indianapolis, IN, USA, October 26-31, 2013*, 2013. ACM, 641–660. https://doi.org/10.1145/2509136.2509549

[4] Sam Blackshear, Bor-Yuh Evan Chang, and Manu Sridharan. 2013. Thresher: Precise Refutations for Heap Reachability. *ACM SIGPLAN Notices* 48, 6 (June 2013), 275–286. https://doi.org/10.1145/2499370.2462186

[5] Amiangshu Bosu, Fang Liu, Danfeng (Daphne) Yao, and Gang Wang. 2017. Collusive Data Leak and More: Large-scale Threat Analysis of Inter-app Communications. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, April 02, 2017. ACM, Abu Dhabi United Arab Emirates, 71–85. https://doi.org/10.1145/3052973.3053004

[6] Kwanghoon Choi and Byeong-Mo Chang. 2014. A Type and Effect System for Activation Flow of Components in Android Programs. *Information Processing Letters* 114, 11 (2014), 620–627. https://doi.org/10.1016/j.ipl.2014.05.011

[7] Anthony Desnos and Geoffroy Gueguen. 2011. Android: From Reversing to Decompilation. *Black Hat Abu Dhabi* (2011). Retrieved from https://media.blackhat.com/bh-ad-11/Desnos/bh-ad-11-DesnosGueguen-Andriod-Reversing_to_Decompilation_WP.pdf

[8] William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol N. Sheth. 2010. TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones. In *9th USENIX Symposium on Operat-*

ing Systems Design and Implementation, October 2010. USENIX Association, Vancouver, BC, Canada, 393–407.

[9] Sascha Fahl, Marian Harbach, Thomas Muders, Lars Baumgärtner, Bernd Freisleben, and Matthew Smith. 2012. Why Eve and Mallory Love Android: An Analysis of Android SSL (in)Security. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, October 16, 2012. ACM, Raleigh North Carolina USA, 50–61. https://doi.org/10.1145/2382196.2382205

[10] Hugo Gascon, Fabian Yamaguchi, Daniel Arp, and Konrad Rieck. 2013. Structural Detection of Android Malware Using Embedded Call Graphs. In *Proceedings of the 2013 ACM Workshop on Artificial Intelligence and Security*, November 04, 2013. ACM, Berlin Germany, 45–54. https://doi.org/10.1145/2517312.2517315

[11] Dimitris Geneiatakis, Igor Nai Fovino, Ioannis Kounelis, and Pasquale Stirparo. 2015. A Permission Verification Approach for Android Mobile Applications. *Computers & Security* 49, (March 2015), 192–205. https://doi.org/10.1016/j.cose.2014.10.005

[12] Michael I. Gordon, Deokhwan Kim, Jeff H. Perkins, Limei Gilham, Nguyen Nguyen, and Martin C. Rinard. 2015. Information Flow Analysis of Android Applications in DroidSafe. In *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015*, 2015. The Internet Society.

[13] Johannes Hoffmann, Martin Ussath, Thorsten Holz, and Michael Spreitzenbarth. 2013. Slicing Droids: Program Slicing for Smali Code. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing (SAC '13)*, March 18, 2013. Association for Computing Machinery, New York, NY, USA, 1844–1851. https://doi.org/10.1145/2480362.2480706

[14] Jinseong Jeon, Kristopher K. Micinski, Jeffrey A. Vaughan, Ari Fogel, Nikhilesh Reddy, Jeffrey S. Foster, and Todd Millstein. 2012. Dr. Android and Mr. Hide: Fine-Grained Permissions in Android Applications. In *Proceedings of the Second ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*, October 19, 2012. ACM, Raleigh North Carolina USA, 3–14. https://doi.org/10.1145/2381934.2381938

[15] William Klieber, Lori Flynn, Amar Bhosale, Limin Jia, and Lujo Bauer. 2014. Android Taint Flow Analysis for App Sets. In *Proceedings of the 3rd ACM SIGPLAN International Workshop on the State of the Art in Java Program Analysis*, June 12, 2014. ACM, Edinburgh United Kingdom, 1–6. https://doi.org/10.1145/2614628.2614633

[16] Li Li, Alexandre Bartel, Tegawende F. Bissyande, Jacques Klein, Yves Le Traon, Steven Arzt, Siegfried Rasthofer, Eric Bodden, Damien Octeau, and Patrick McDaniel. 2015. IccTA: Detecting Inter-Component Privacy Leaks in Android Apps. In *2015 IEEE/ACM*

37th IEEE International Conference on Software Engineering, May 2015. IEEE, Florence, Italy, 280–291. https://doi.org/10.1109/ICSE.2015.48

[17] Li Li, Alexandre Bartel, Tegawendé F. Bissyandé, Jacques Klein, and Yves Le Traon. 2015. ApkCombiner: Combining Multiple Android Apps to Support Inter-App Analysis. In *ICT Systems Security and Privacy Protection*, 2015. Springer International Publishing, Cham, 513–527. https://doi.org/10.1007/978-3-319-18467-8_34

[18] Li Li, Tegawendé F. Bissyandé, Mike Papadakis, Siegfried Rasthofer, Alexandre Bartel, Damien Octeau, Jacques Klein, and Yves Le Traon. 2017. Static Analysis of Android Apps: A Systematic Literature Review. *Information and Software Technology* 88, (2017), 67–95. https://doi.org/10.1016/j.infsof.2017.04.001

[19] Shuying Liang, Andrew W. Keep, Matthew Might, Steven Lyde, Thomas Gilray, Petey Aldous, and David Van Horn. 2013. Sound and Precise Malware Analysis for Android via Pushdown Reachability and Entry-Point Saturation. In *Proceedings of the Third ACM Workshop on Security and Privacy in Smartphones & Mobile Devices (SPSM '13)*, November 08, 2013. Association for Computing Machinery, New York, NY, USA, 21–32. https://doi.org/10.1145/2516760.2516769

[20] Max Lillack, Christian Kästner, and Eric Bodden. 2014. Tracking Load-Time Configuration Options. In *Proceedings of the 29th ACM/IEEE International Conference on Automated Software Engineering (ASE '14)*, September 15, 2014. Association for Computing Machinery, New York, NY, USA, 445–456. https://doi.org/10.1145/2642937.2643001

[21] Yepang Liu, Chang Xu, and Shing-Chi Cheung. 2014. Characterizing and Detecting Performance Bugs for Smartphone Applications. In *Proceedings of the 36th International Conference on Software Engineering*, May 31, 2014. ACM, Hyderabad India, 1013–1024. https://doi.org/10.1145/2568225.2568229

[22] Linghui Luo, Felix Pauck, Goran Piskachev, Manuel Benz, Ivan Pashchenko, Martin Mory, Eric Bodden, Ben Hermann, and Fabio Massacci. 2022. TaintBench: Automatic Real-World Malware Benchmarking of Android Taint Analyses. *Empirical Software Engineering* 27, 1 (January 2022), 16. https://doi.org/10.1007/s10664-021-10013-5

[23] Damien Octeau, Daniel Luchaup, Matthew Dering, Somesh Jha, and Patrick McDaniel. 2015. Composite Constant Propagation: Application to Android Inter-Component Communication Analysis. In *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, May 2015. IEEE, Florence, Italy, 77–88. https://doi.org/10.1109/ICSE.2015.30

[24] Damien Octeau, Patrick McDaniel, Somesh Jha, Alexandre Bartel, Eric Bodden, Jacques Klein, and Yves Le Traon. 2013. Effective Inter-Component communication mapping in

android: An essential step towards holistic security analysis. In *22nd USENIX Security Symposium (USENIX Security 13)*, 2013. 543–558.

[25] Felix Pauck, Eric Bodden, and Heike Wehrheim. 2018. Do Android Taint Analysis Tools Keep Their Promises?. In *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, October 26, 2018. ACM, Lake Buena Vista FL USA, 331–341. https://doi.org/10.1145/3236024.3236029

[26] Feargus Pendlebury, Fabio Pierazzi, Roberto Jordaney, Johannes Kinder, and Lorenzo Cavallaro. 2018. TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time. (2018).

[27] Sebastian Poeplau, Yanick Fratantonio, Antonio Bianchi, Christopher Kruegel, and Giovanni Vigna. 2014. Execute This! Analyzing Unsafe and Malicious Dynamic Code Loading in Android Applications. In *21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, California, USA, February 23-26, 2014*, 2014. The Internet Society.

[28] Bradley Reaves, Jasmine Bowers, Sigmund Albert Gorski III, Olabode Anise, Rahul Bobhate, Raymond Cho, Hiranava Das, Sharique Hussain, Hamza Karachiwala, Nolen Scaife, Byron Wright, Kevin Butler, William Enck, and Patrick Traynor. 2016. *droid: Assessment and Evaluation of Android Application Analysis Tools. *ACM Computing Surveys* 49, 3 (October 2016), 1–30. https://doi.org/10.1145/2996358

[29] Atanas Rountev and Dacong Yan. 2014. Static Reference Analysis for GUI Objects in Android Software. In *Proceedings of Annual IEEE/ACM International Symposium on Code Generation and Optimization*, February 15, 2014. ACM, Orlando FL USA, 143–153. https://doi.org/10.1145/2544137.2544159

[30] Feng Shen, Namita Vishnubhotla, Chirag Todarka, Mohit Arora, Babu Dhandapani, Eric John Lehner, Steven Y. Ko, and Lukasz Ziarek. 2014. Information Flows as a Permission Mechanism. In *Proceedings of the 29th ACM/IEEE International Conference on Automated Software Engineering*, September 15, 2014. ACM, Vasteras Sweden, 515–526. https://doi.org/10.1145/2642937.2643018

[31] Dennis Titze and Julian Schutte. 2015. Apparecium: Revealing Data Flows in Android Applications. In *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*, March 2015. IEEE, Gwangiu, South Korea, 579–586. https://doi.org/10.1109/AINA.2015.239

[32] Timothy Vidas, Jiaqi Tan, Jay Nahata, Chaur Lih Tan, Nicolas Christin, and Patrick Tague. 2014. A5: Automated Analysis of Adversarial Android Applications. In *Proceedings*

of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices, November 07, 2014. ACM, Scottsdale Arizona USA, 39–50. https://doi.org/10.1145/2666620.2666630

[33] Fengguo Wei, Sankardas Roy, Xinming Ou, and Robby. 2014. Amandroid: A Precise and General Inter-component Data Flow Analysis Framework for Security Vetting of Android Apps. In *ACM SIGSAC Conference on Computer and Communications Security*, November 2014. ACM, Scottsdale Arizona USA, 1329–1341. https://doi.org/10.1145/2660267.2660357

[34] Erik Ramsgaard Wognsen, Henrik Søndberg Karlsen, Mads Chr. Olesen, and René Rydhof Hansen. 2014. Formalisation and Analysis of Dalvik Bytecode. *Science of Computer Programming* 92, (October 2014), 25–55. https://doi.org/10.1016/j.scico.2013.11.037

[35] Mingyuan Xia, Lu Gong, Yuanhao Lyu, Zhengwei Qi, and Xue Liu. 2015. Effective Real-Time Android Application Auditing. In *2015 IEEE Symposium on Security and Privacy*, May 2015. IEEE, San Jose, CA, 899–914. https://doi.org/10.1109/SP.2015.60

[36] Shengqian Yang, Dacong Yan, Haowei Wu, Yan Wang, and Atanas Rountev. 2015. Static Control-Flow Analysis of User-Driven Callbacks in Android Applications. In *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, May 2015. IEEE, Florence, Italy, 89–99. https://doi.org/10.1109/ICSE.2015.31

[37] Yury Zhauniarovich, Maqsood Ahmad, Olga Gadyatskaya, Bruno Crispo, and Fabio Massacci. 2015. StaDynA: Addressing the Problem of Dynamic Code Updates in the Security Analysis of Android Applications. In *Proceedings of the 5th ACM Conference on Data and Application Security and Privacy*, March 02, 2015. ACM, San Antonio Texas USA, 37–48. https://doi.org/10.1145/2699026.2699105

COLLEGE DOCTORAL BRETAGNE
MATHS, TELECOMS
INFORMATIQUE, SIGNAL
SYSTEMES, ELECTRONIQUE

CentraleSupélec

Titre : TODO 18 ▶ ***Find a title*** ◀

Mots clés : Android, Analyse de Maliciels

Résumé : Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque.

Title : TODO 19 ▶ ***Find a title*** ◀

Keywords: Android, Malware Analysis, TODO 20 ▶ ***More Keywords*** ◀

Abstract: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque.