National College of Ireland

Augmenting Privacy through Metadata Wiping

MSc Research Project

Cybersecurity

Simon Lowry

Student ID: x21168938

School of Computing

National College of Ireland

Supervisor: Michael Pantridge

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Simon Lowry
Student ID:	x21168938
Programme:	MSCCYBE_JANOL_O **Year** 2 of 2
Module:	Research Project
Supervisor:	Michael Pantridge
Submission Due Date:	01/12/2023
Project Title:	Augmenting Privacy Through Metadata Wiping
Word Count:	8943 words **Page Count:** 20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: *Simon Lowry*

Date: 02/12/2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Augmenting Privacy through Metadata Wiping

Simon Lowry

x21168938

Abstract

Metadata has become increasingly prevalent with the explosion of big data. The effects of its collation have meagerly been brought to public discourse and is capitalized upon by those know how. The lack of public understanding creates a numb disinterest in metadata. This asymmetry creates a breathing ground for exploit with those sufficient knowledge. Metadata and it's spread of harms has been direly under researched. Tools created have often been limited in scope with considerations sometimes to the detriment of their very intent to provide increased security. This paper and tool has sought to rectify these issues. It looks to provide a universally accessible tool to cover multiple file types to help sustain privacy for individuals and organisations alike. This will help in protecting intellectual property, sensitive information whether it be PII, PHI or litigation documents or classified information which can persist in files. It shows the dangers of why tools like this are becoming increasingly important in a world where passive and exposed surveillance have become more and more all-encompassing. It exposes the asymmetry that exists between those who know how to make use of this information and how that's being exploited. It identifies where other current tools and state of the art research papers have been lacking and proposes a solution which improves on current work. It aims to increase the awareness of why it matters to care about metadata.

1 Introduction

Metadata is often the unintended byproduct of our files, interactions and actions online. It's part of our digital footprint that to the uninitiated may seem closer to exhaust fumes of our actions. That's not the case for those who know how to exploit it. In the words of the former director of the NSA General Michael Hayden, "We kill people based on metadata". Another of his NSA colleagues describes it "Metadata absolutely tells you everything about somebody's life. If you have enough metadata, you don't really need context". [8]

Privacy of the average individual is increasingly becoming deprived with the ubiquitous collection and collating of metadata obtained from various sources including file metadata. It can not only identify individuals, their behaviour online but also means of targeting them. Even the privacy of our pets' location (and by extension our own) is not safe. As demonstrated by Owen Mundy, he was able to present a map of where cats live based on the metadata acquired from photos. This may seem innocuous but the implication here is that their owners location can be obtained from their photos too.This could be a family member, a colleague or anyone you know. This experiment opens the door on what's to come and highlights why it matters to care about the metadata we often unknowingly and casually give away. [14] Our images can tell nefarious actors the type of camera, the model, the location and facial recognition metadata & these properties can also be dangerous as well as expanded on later.

An informal definition of metadata would be something along the lines of information about an object or process. It's often recorded and collected as a way of identifying trends, describing the data enclosed in a given file, modelling potential scenarios and for administering different algorithmic solutions. The main categories

include descriptive (centered around identification details), structural (details about the containers) as well as administrative (which are more to do with creation, access and technical details). Structural metadata is less relevant here as it doesn't contain the sensitive information disclosures which other types can like descriptive and administrative.

With the explosion of big data and expanded masses of files making their way online, the metadata enclosed on these files becomes a greater threat vector. IOT also brings in another swathe of devices into our very homes which again can have metadata associated with them whether that be in the form of serial numbers for devices. Devices which are notoriously not designed with security in mind, often cheaply made and not updated. Fitbit and strava have also inadvertently disclosed the location of certain military bases. [9]

A high severity CVE (Common Vulnerability and Exposure) identified in 2019 outlined how a poorly designed verification check could well be exploited in relation to serial numbers obtained through metadata. This vulnerability is exposed during the associating process which occurs with a new camera being set up to associate with Amcrest cloud services. During the process the serial number if obtained and a few conditions met, like a reset in the last two hours, the camera can then be completely controlled by an attacker. This includes viewing what the camera sees, listening in on the audio from the microphone. This clearly showcases a massive breach of privacy and all of this starts from the metadata acquired which in this case is the serial number. [10] [4]
On top of that, obtaining that information or assessing metadata with photos of a given individual's home can serve as a location indicator for any thieves casing a location. A precursor employed by thieves to understand the layout and what they might want to rob before they go and attempt that robbery..

Other threats posed by exposed metadata include social engineering and cyber exploitation success can be compounded by the potent effect of specific metadata. Attackers may look at gaining and harvesting information such as email address, phone numbers or affiliations. Spear phishing with targeted and tailored campaigns can leverage personal information that can be used against individuals to exploit them. Identity theft could also be possible. Drive-by downloads, watering hole attacks and man in the middle attacks are all given more valence and likelihood of success with relevant metadata acquired. [13] Stalking and harassing can occur here too. Metadata is also becoming a key component in online surveillance and is being made available to unvetted and unregulated third parties by even ISPs and dragnet surveillance gorging on big data. This threatens those in countries with large amounts of surveillance to an even greater degree. A document with information that is not allowed in one country and allowed in most others may become a means of revealing an individual's identity. This can happen through the metadata of the file. This is very broadly the case with exposed documents and other file types which can reveal sensitive information.

Lawyers can also be at the mercy of poor handling of metadata. When conducting the sharing of documents with opposing litigation teams, poor practices can accidentally result in revealing privileged and confidential information. [11] This could be equally applied in a business setting leaving documents exposed on a not properly restricted repo or also IP information contained with the metadata of those files. The same for classified information in relation to the military. Another example related to a report produced on the Iraq war, where the authors inadvertently left their names in the metadata of the report, leaving these individuals facing unwanted public scrutiny. [12]

Inferences can be made from certain metadata attributes, they need not be deductive to be useful. As such with other pieces of information it can make the incomplete leaps afforded to a higher degree of certainty of for example a document with only a created and modified date, being aligned with a whole other set of documents of the same nature. This highlights the asymmetry of knowledge of those with sufficient technical or otherwise knowledge. Those with the technical capabilities at their disposal can take advantage of this information. While in contrast, the average individual may consider it benign and useless. [28] That casual approach elevates it as a tool for exploitation by those with the means and motivation since it's less likely to be stripped or encrypted. Thus the responsibility falls somewhat on security researchers to raise awareness. It's also necessary to provide freely available tools like this one to help those who want to protect themselves. The more papers on this topic

the more likelihood it can provoke further discussion. This could then lead to greater legislation that may be required to protect the privacy and sovereignty of individuals and companies as well when it comes to uploading files. [3]

While GDPR has helped in protecting some personal information it appears that policy and laws are still a step behind in terms of enforcing the protection of our metadata. There's too great an incentive for corporations and government agencies alike to keep it that way. The selling of our metadata can be a lucrative business, a means of maintaining control for more invasive governments and a way to target individuals for agencies like the NSA. As a result of these reasons and more, legislation still lags behind in this area.

There's a slippery slope from countries declining towards greater surveillance and decreased freedoms of individuals. With that and greater restrictions on privacy, there's also the erosion of the freedom of the press. Even in Australia, police have been able to leverage metadata to arrest whistleblowers who provided journalists with embarrassing materials related to government officials. Refugees were subject to abuse and brutality and government officials were complicit in this, this was brought to journalists and published and arrests followed for those whistleblowers. On top of this a law was passed in Australia which obligates ISPs to maintain metadata of all traffic. Even in Australia, police have been able to obtain all of this metadata to arrest whistleblowers. A journalist who writes for the guardian who was targeted by this very law, Ferell, explained that this has not been for the sake of national security but instead about preventing uncomfortable truths about those in power, coming to light. [24]

In China, a blanket ban was placed on over 50,000 multimedia files. 57% of those files were common religious materials including chapters from the Quran. Cultural and peaceful expressions of Uyghurs and other Turkic Muslims has been unjustly conflated with terrorism. These people's ability to read and enjoy texts which have great significance to their lives is being forcibly taken away from them. Individuals found with deemed offensive material are interrogated, detained in political "education" camps or sentenced to prison without any legal representatives or open trials. An estimated half a million people remain imprisoned on the back of this crackdown. [29] Dystopian outcomes like the ones mentioned are happening incrementally more and more and are becoming increasingly worse in the background of most people's lives. The chipping away at democracies, freedom of press, abuses of power are all occurring. Same for limitations of freedoms generally and unjust treatment and this is shown to be carried out through the use of metadata. This necessitates increased measures to protect one's privacy and choices around metadata.

Having tools which can strip metadata of our multitude of file types is becoming increasingly important. A tool designed with regular users in mind and not programmers. The vast majority of tools available are limited in scope and are often command line tools. This narrows the scope of their effectiveness and misses the vast majority who do not code. More on this later in the lit review and related work section.

Motivation

The main motivations for this tool and paper include a user friendly approach to stripping metadata on multiple different file types. This will be done with a web application usable by those without programming knowledge in contrast to some other tools in the space. Another key motivation is to raise awareness of the dangers that can be inherent with a lackadaisical approach to our file metadata.

The pros and cons of these various tools have all been taken into consideration and played a substantial role in how this application has been designed. Metadata Wiper is a novel tool that has been designed to cover a number of the most ubiquitously used file types which contain metadata. It's not just for jpegs for example like some of the other tools mentioned. It thus offers a great opportunity for users to sustain the confidentiality of their own data across that breadth of file types. It's a web application which is expanded on in the related work section.

This tool is about eradicating metadata from an array of different file types. If you want to individually change or modify metadata on a given file, other tools are already available for that. A lot of the tools are set for one

individual format and are focused on different tasks than this tool. Here, this tool is created for a clear purpose in privacy and protection through removal of data associated with files. There's a vast spread of use cases for this tool. One such example could be an individual who wants to ensure the metadata for this file or set of files is reduced as much as possible. This could be someone in a country where their privacy is impinged on by governments whether it be a journalistic capacity or just a regular citizen. In countries where freedom of the press or dissenting views of the government can result in going to prison, metadata could very well betray an individual. The very metadata attached to journalism or expressing opinions counter to those in power could end up being used against them. Even when changing file format, for example from docx to pdf, metadata can still be sustained. [7] Metadata can also be revealed in the process of downgrading sensitive confidential material in military contexts. While the regular procedure is to redact sensitive information, metadata can be forgotten and be a source of information leakage.

Another use case could be a company that does not want the wrong people to gain access to the metadata of certain files and thus, use it to wipe the metadata provided. It may well help against GDPR claims as well with reducing the metadata of customers and clients alike. PII or PHI data or unique identifiers could be compromised. This could be beneficial in the events of a breach of customer data if the would be attackers are not able to determine any individuals associated with the given files. It reduces their ability to be able to blackmail individuals since they effectively won't know who it belonged to or when it was created or modified by.You can't target an individual with photos for example if you don't know who that photo is of. Metadata from these files is prevented from being involved in social engineering attacks or spear phishing attacks as well. These can combine metadata from a multitude of sources to try and trick someone and exploit.

On top of the above use cases, a number of tools can tend to append and add metadata to files that result in situations unfolding that need not have occurred otherwise. This might be location data or otherwise appended to a photo without the individual's realisation. Retaining the privacy of documents such as a religious text which can be very meaningful to someone ought to be kept in their possession at their own discretion. The same applies to intellectual property or journalists looking to protect their sources. Keeping individuals' privacy more in their own hands instead of at the disposal of the technology they're using or at the disposal of those who might abuse that information. Individuals and groups who can exploit that information with an asymmetry of understanding of how it can be exploited and a public which hasn't been fully exposed to where metadata can be a danger and thus, take it for granted. This research paper and tool looks to assist in regaining people's privacy.

2 Related Work & Literature Review

In this section, an analysis of related work is conducted as well as a discussion about related tools currently available. Each paper is critically evaluated and provides the basis for the current paper to expand on. One such paper was by Henne et al. It examined the privacy threats posed by embedded metadata which people may or may not be consciously aware of. It offers a solution through a tool called SnapMe. This paper carried out some really interesting surveys and analysis of the threats of metadata through imagery uploaded online and to various different social media. However, the solution offered up compromising security and privacy in a multitude of ways. Some of the demands of this service are pronounced. An always running service which could potentially have a lot of privilege and ironically trust required to examine all of our media on uploads. They also propose a centralised authority for all photos being taken by all users. The tool proposes selecting co-ordinates (also known as geo-fencing) where you will get a feed of images that have been taken in that area. What is stopping a stalker from setting an area of their choosing and getting all of the available images? Or mass surveillance getting a vast amount of images from areas they set? On top of that, if a user forgets to stop their dynamic area of collecting this data, they are exposing themselves completely. This would also be a tool exploited by stalkers & for mass surveillance in a heartbeat. Any attacker exploits that system likewise has the same trove of data. There is no mention of how this data will be protected, how long it would be stored, if it would be deleted, or would the users be able to request its even deletion and how that would take place. [3]

It set off with the ambition of being a privacy tool and seems to have undermined that very cause through its design. There's a clear lack of threat modelling, abuse cases and other security requirements to think about what could go wrong here and how it could be exploited or abused. No mention of how they are securing the data or would look to adhere to GDPR or any other protections for PII data. Too much trust expected and too much work expected of the user in terms of maintenance. This reduces its usability and increasing the likelihood of inadvertent security mishaps.

The system creators want to be able to assess every single piece of media on our phones, this is the antithesis of privacy and security. When the NSA had access to cameras and devices of people this was abused as revealed by former NSA contractor and whistleblower Edward Snowden. [27] On top of this, Social media websites have been exploiting the use of this metadata and imagery to make profits for the last two decades. What would stop that happening here? Any government that puts pressure on the owners of a system like this gains the location of every user. Any hacker could gain the same throve of personal data. There is no accounting for anything like GDPR or regulations or anything of the sort. Little mention of real safeguards, instead the attitude of: just trust us. This paper helped inform the proposed solution of the current paper. Ir will look to minimize its maintenance of data obtained from the user. The files which are being obtained will instantly be deleted. This reduces the load of storage for the application and prevents the files becoming toxic storage. There is no justifiable reason to maintain those files beyond that. [3]

Another paper in this area is securing image metadata using AES. However, there is little provided as to why this matters. Why it's important to secure metadata and the motivation is not expanded on. The use case seems to be implied rather than clearly stated. In certain legal, forensic or copyright proceedings it may be beneficial to maintain the metadata while still maintaining its confidentiality. This may have been potentially a motivation or use case for a tool like this instead of wiping the data as we are doing in this paper. There are a number of cryptographic algorithms listed as well as some cryptography basics and how AES works. A brief description of some generic strengths and weaknesses without going into much exploration on the comparison of these algorithms. There doesn't seem to be any in depth analysis performed on the approach taken for the system designed. The analysis and research conducted appears to be shallow. Security considerations are again not mentioned for the design of the application. There are no steps to assess whether the file is malicious. Nothing is done to protect against any malicious input. This paper helped highlight the importance of clearly stating the motivations and what it's important about this current paper. Again we also see the importance of rigour in the considerations for the application itself and having a secure design. [1]

Another paper which looked at removing metadata as part of an upload process for social media users. They focus again purely on images and only on Jpeg images. This leaves a number of other primary core file types which are widely used unexplored and also potentially revealing sensitive information. The application proposed is a desktop application, not available on mobile and not OS independent. This paper is however more explicit in its aims, justifications and provides some contextual information about metadata giving the reader more insight. There are some false suggestions that metadata can be manipulated directly on these websites which is not correct, it can be read but not modified directly on the website. There aren't many examples though on the dangers that can occur from not wiping metadata and why it's a pressing concern. [6]

Metadata is one important piece of the puzzle where information leaks, PHI or PII can be revealed. A paper by Tuomas Aura et al highlights that there are a number of other ways in which PII and other data can be leaked inadvertently. This ranges from poor redacting, to thumbnails embedded in image objects. It also includes GUIDs which identify unique network interfaces, anonymised conference submissions and strings spread throughout different parts of documents accidentally left in there or added by different pieces of software. Human generated comments, machine generated comments, hyperlinks can reveal PII or internal company websites too. This paper effectively shows where some of the barriers and limits of this research will conclude. These other aspects are covered by other tools, more on this in the limitations section. [7]

More relevant research was covered on privacy in big data which was centered around metadata as well by Smith et al. The paper highlighted the huge quantities of big data and metadata along with it being posted online. The number of photos posted on facebook rose from 2 billion to a staggering 6 billion per month.[25] [26] The authors acknowledged that the lack of control of other users posting images about us is becoming a greater problem. The metadata was often also maintained in images for example. Flickr was taken as a case study example. They analysed over 20,000 photos from users and were able to find GPS locations of users present on 34% of those photos alone. These were all GPS available. There was a clear dominance of those with gps data coming from mobile photos. Beyond the GPS location, there was also camera ids, street and city locations as well as PII information. This is a trend to watch as well, as camera ids, street and city locations are often not stripped from those who do remove metadata, these can still be obtained and exploited. [2]

Again, the authors propose a tool that's using GPS to have suggested privacy zones. In these privacy zones the user is informed about media uploaded by others to social media sites and other sharing sites. The nefarious use case is not considered for less ethical types or people looking to exploit or even stalk others could abuse this kind of technology. It would undermine the very attempt to achieve privacy as mentioned with a previous paper

that had a similar idea. A study conducted by Goeurt et al found that of the 33 popular web services (including social media sites and forums) which allowed uploading of images, nearly a third of them do not perform any kind of sanitization. [4] Therefore, private information about those users and their devices is available to any would-be attacker and people are not being sufficiently protected by these web sites so it's clear that we are not able to purely rely on these various sites to do this for us. They do not face enough scrutiny for these aspects of privacy being breached as of yet. [2]

Another part of the related research was conducted on papers and tools that are of a similar nature. Toevs showed in his research the use and information about Exiftool and its application. [5] It's a popular tool that can modify and delete metadata on images. Its primary way of doing this is by command line tool or API. That's helpful for users who have programming knowledge and capabilities but doesn't help as much, the rest of the people on the planet. MetadataWiper is a web application so that it's usable by the widest audience possible. According to developer nation, there's only 0.003% of people on the planet who are programmers. This is 25.8 million of 7.888 billion. The remainder of the 99.007% of the planet could benefit from a tool as well. Tools like Exif require programming knowledge as it's command line based. That leaves an abundant market share which has not been tapped into and alienates non technical users from its use. Producing a web application here corrects that. It makes it possible to be available for anyone with a browser. It's also OS independent and deployable and usable on mobile as well. Other tools were also examined: ScrubDoc, WordMetadataChanger, Clean Docs & ExifEraser. What they have in common is that they act on only a single type of file. The first three on word doc files and the last one on images. MetadataWiper is focused on four different file types which are ubiquitously used throughout the globe. Those four types are JPG, XLSX (excel files), docx (word files) & PDF. While JPG has received some attention and research rightly for location information, these other file types have received a lot less attention and can equally pose risks to an individual or companies privacy. Providing multiple file types gives greater control for removing unwanted metadata across these important file types.

3 Research Methodology & Design

Overall, the literature review informed this research paper in a number of ways. It provided a spread of approaches and alternative methodologies to glean insight from and see where they were lacking or could be improved on. This critical examination shows the gap for a metadata wiping tool which has multiple formats available. That became clear and the use cases were abundant. The necessity for providing numerous file types with the capacity to be wiped is a must. Their metadata can equally be as disastrous when exploited and/or combined with other information. Research and tools have been sorely lacking beyond JPG files. It shows the gap there is currently for a tool which is easily accessible and usable across all platforms and not just available to a minority of folks who can program. While this may not seem like a massively complicated aspect of the tool it's still very important. Bringing availability of tools like this from a miniscule percentage of people to the biggest audience available through a web application gives anyone with a browser a chance to gain more privacy. It showed that operating with a shift left mindset of incorporating security from requirements, design and onwards is needed. This helps to produce a tool which assists increasing privacy and security of individuals and companies alike, while also not undermining that very effort. Some of the papers and approaches employed ineffective security design and threat modelling that undermined the very privacy they were looking to obtain. Attempting to think like a hacker from the outset can help uncover security issues in the design phases and add in mitigations from the get go. Rigour and due diligence is required for all aspects of the creation as well and this was played out from the design phase onwards which is explored in the next section and this paper has benefitted from being able to see some of the pitfalls that have come before it. It also highlighted the need for outlining a clear motivation, use cases and benefits of the proposed tool which has been included. All of these alternative research methodologies and directions explored have paved the way for what's included in this paper and where the next incremental improvements could be made.

A. Materials

Python was deemed the most popular programming language by the TIOBE index in 2022 when it surpassed Java. [2] It's established itself as the de facto programming language of cybersecurity professionals providing a collection of libraries tailored specifically towards this industry. It's also an open source language and gains the benefits from that having a vast community of developers providing necessary scrutiny to its codebase. Its latest version is 3.11 which was released in 2022. It's updated often and supports multiple platforms.

Veraode conducted a study which explored the security vulnerabilities found in applications by programming languages. [17] [18] It explored applications in the following languages: C#, Python, Java, Javascript and Php, some of the used programming languages. In their research around 130,000 different applications were scanned.

The report established that 74% of applications were found to have at least one security vulnerability. It also performed some comparative analysis among the languages and then a deep dive into languages specifically including Python.

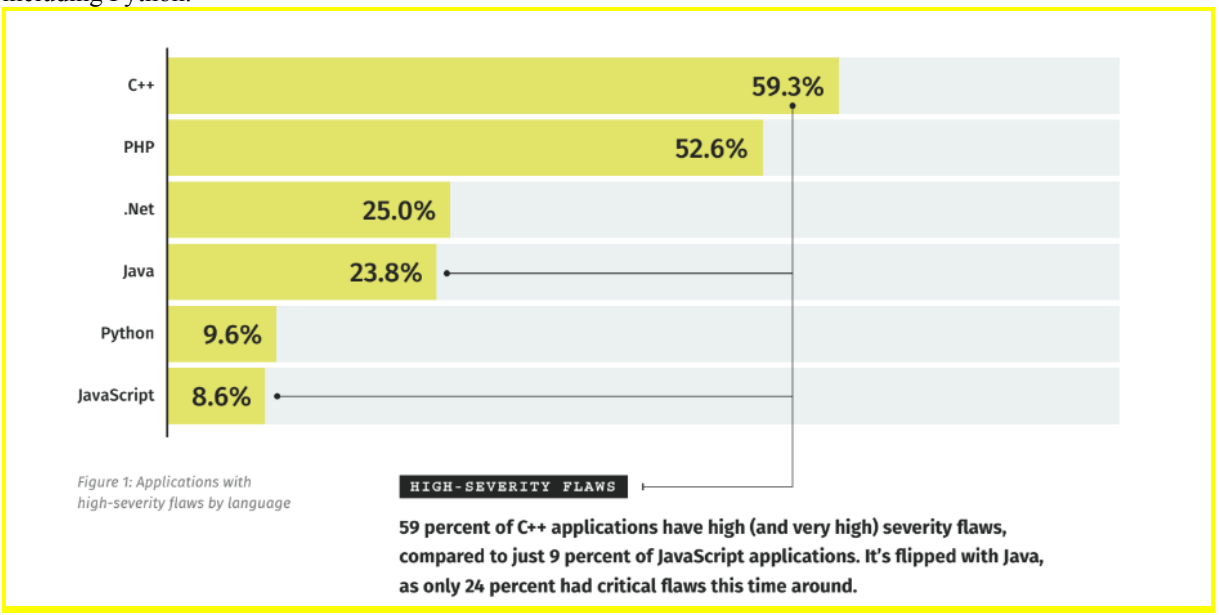

Fig 3. 1 [18]

The Veracode analysis showed that Python applications have significantly less high severity security flaws coming in at 9.6%, this is hugely lower than other popular languages like C++ at 57.3%, Php at 52.6% and java at 23.8%. Only javascript bettered Python in this study. This acts in Python's favour for selection as the programming language of choice in this project. On top of that, Python tends to be language of choice for cyber security practitioners. However, the language's libraries are not without their own issues which require some steps to ensure security and that'll be discussed next.

Pythons packages can be installed via Pip a command line with it's packages being stored in Pypi. A study on pypi distribution packages showed that 46% of the libraries present contain a security vulnerability in them. [5] This places the onus squarely on us developers to ensure due diligence has been performed on the libraries being used from there. As, there is no process of rigour in place to reject packages with the spread of vulnerabilities in them

An important way to mitigate against this is to run Python in a virtualized environment. Thus, if a given package is vulnerable, this reduces the likely impact and protects other projects and the main host environment. Django does also offer a way of doing this itself and this is expanded on in the configuration manual. Selecting the packages for download is another aspect which requires careful consideration. For example, a given package with the name "00Seven" is completely different from another package "000Seven" even though their names are remarkably similar. With these considerations taken into account, the packages used in this project have been investigated for security issues. This is necessary with vulnerable components becoming a greater risk and even featuring on the latest OWASP Top 10, at number 6 on the list.

Some other challenges related to security when using python are to do with importing packages, string formatting and http requests which will be dissected next. The project will be making use of Django, RE, os, http requests as a web application and string formatting may come into play too. The analysis of these tools and features has informed the choices around their selection how to make use of these features and libraries in a secure fashion.

There are three ways of importing packages in Python. The first is absolute, there is also the implicit and relative imports. Setting up implicit paths can be done without requiring the exact location of the package. From this, trojan horses or malicious packages can look to pretend to be the real package but instead contain malicious code. Opting for absolute paths instead of implicit paths can ensure the risks here are mitigated and the files are protected against this. Django is going to be an important framework in this application and has some risks associated with it. Setting debug to true is activated by default with Django. As a result of this errors can be output and give attackers information which can help them to know where the application is vulnerable. [21]

Another consideration for this application is string formatting which poses its own risks. Python 3 has introduced some formatting within f-string and str.format() that have been shown to leak sensitive data. A more secure approach can be instead to go with the use of the Template class which is a part of the String class. It's useful for both generated data as well as user provided input. [19] The use of string formatting can be in play with protecting against a spread of different injection attacks whether it be XSS or SQL injection or others. The library RE is also helpful here, which stands for regular expression and describes its functionality. It makes it easy to create allow lists of characters with the format of regular expressions. [23] Beyond that there is also a library called os which will be used in assessing and validating file details and identifying the size of files as part of security control that will be in place related to files. There's also the security considerations for performing http requests. Requests is a python library that's prominent in this space. However, earlier versions of this library have known vulnerabilities so choosing the most up to date version is important.

C. Threat Modeling and Abuse cases

The Veracode research and examination of python comes into play here as well when threat modelling. Insight into the most prevalent security vulnerabilities in python can be invaluable for narrowing focus and prioritising certain vulnerabilities. These serve as a good starting point in the threat modelling for this application. The top three were Cryptographic issues, which was found in 35% of applications that were scanned, then Cross Site Scripting, found in 22.2% and then finally Directory Traversal coming in at 20.6%. For this application Cryptographic failures could appear in the form of using broken cryptographic algorithms, certificate issues and leaving sensitive information available in clear text. Opting for TLS at it's latest version will help here for data in transit. Certs will only be used under the guise of being a non-production project and as such will not be issued by certificate authority. This would be changed for a production system. Property files will not be included in the repo.

Analyzing the language was just the beginning here and served up relevant potent threats as a beginning of the threat modelling. From there the application specific threats and abuse cases were assessed. Cross-Site Scripting is a form of injection attack whereby the attacker leverages client scripts that are malicious in nature to exploit a web application. It routinely features in the OWASP Top 10, currently at number 3 as part of Injection attacks. There are also a number of CWE references for XSS (CWE-79, CWE-352 & CWE-113) and it's included in the SANS Top 25 Most Dangerous Programming Errors.

Of the three XSS types, which are reflected XSS, stored/persistent XSS and DOM based XSS, only reflected XSS is relevant to this application. It will take in untrusted input provided by the users and then display that information on the screen. Stored XSS and DOM based XSS are not relevant here. Directory traversal is where attackers are able to gain access to restricted directories and files. They were found in almost half (47.8%) of applications in the Veracode report. By applying secure principles such as deny by default this can help here. Given users ought to be able to only access paths based on their privileges delegated to them and nothing more to adhere to least privilege as well. It can also help to block certain user input for paths such as ../ for example and also keeping dependencies up to date and running frequent static analysis scans which are able to identify these vulnerabilities.

Abuse Case

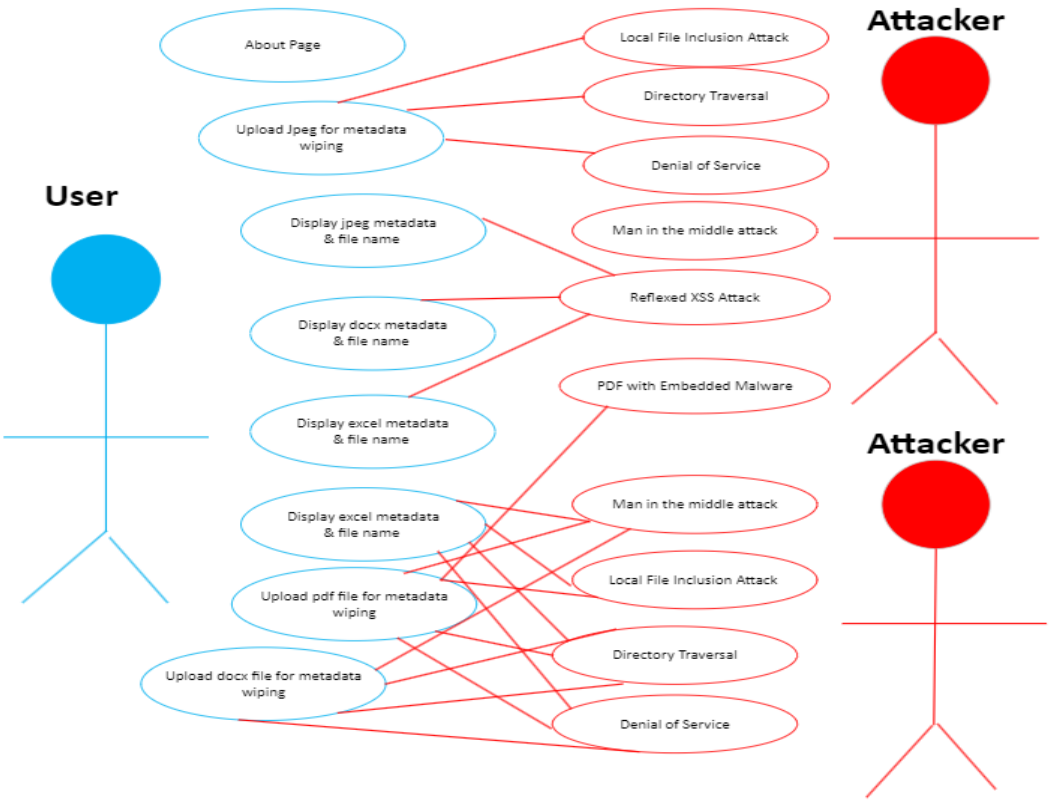

Fig 3.2 - Abuse Cases

Threat Modelling with Mitigations

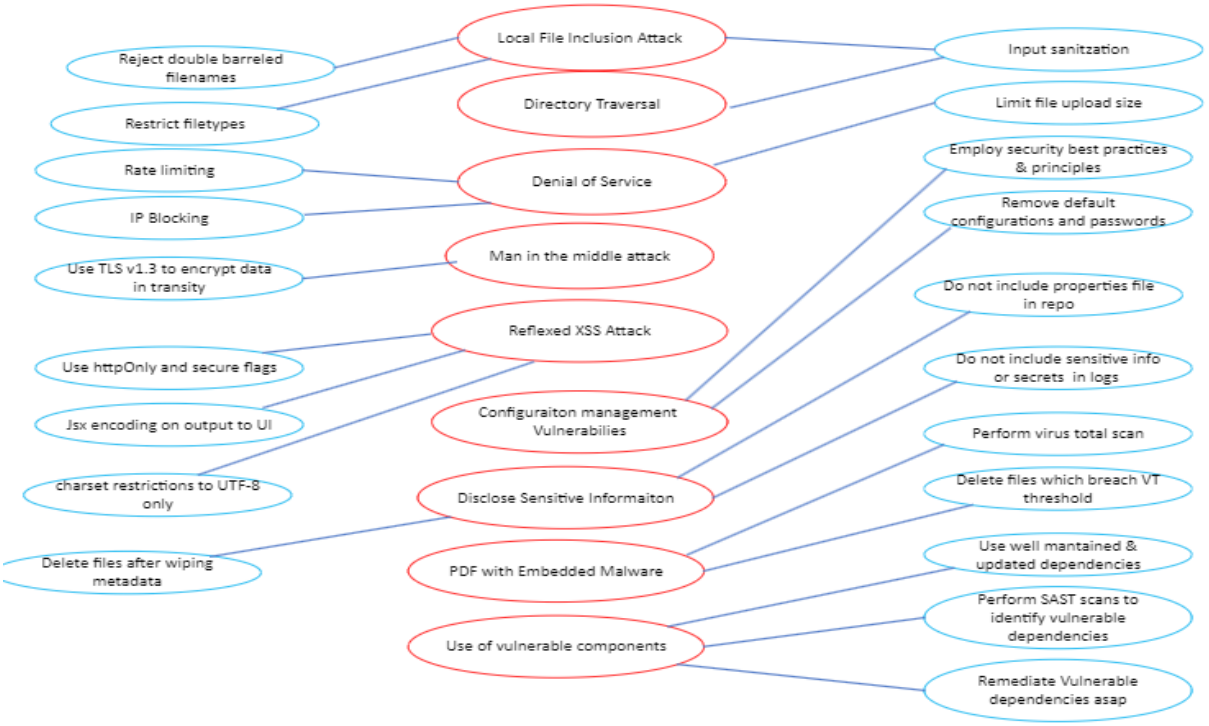

Fig 3.3 - Threat Modelling

From the broader scope and considerations of what the python language and its constituent libraries brought, the threat modelling moved into more specific aspects of the application itself. The application is designed to reduce the attack surface as much as possible. Economy of mechanism is applied to combat against unnecessary complexity, to reduce the likelihood of security vulnerabilities and to limit the attack surface.No unnecessary bloat of features has been added for the sake of it. There was no incentive to have an authentication for this system. It wouldn't bring any functional benefit and would come with broadening the attack surface and potential areas which vulnerabilities could arise as well as increased measures required to protect any sensitive data acquired. This didn't make sense from a functional or security perspective so it wasn't included.

All API's perform input sanitization before taking any further actions. This is carried out on filenames, file types and file sizes. This helps mitigate against cross site scripting, local file inclusion attacks and directory traversal attacks. File sizes have been limited to provide some protection against denial of service. All files are scanned via VirusTotal API before initiating any metadata wiping. If any file has been found to have 5 hits or more on virustotal it's immediately deleted, to protect against any possible malware. No files are kept on the server after performing wiping to prevent information disclosure, they are instead immediately deleted. The properties file is not included in the repo and stopped from inclusion via the gitignore file. Security tools in the form of SAST (Snyk) and DAST (OWASP ZAP) have been used to compound the security efforts as well throughout the implementation..

3 Implementation, Results & Discussion

Firstly in this section, we'll examine the algorithm employed with pseudocode followed by the algorithm displayed as a flow chart. This algorithm was uniformly applied across the various API. Then, we look at the properties of the files which are to be wiped by our metadata wiper tool. This is followed up by the results of performing the wiping functionality on some sample files of the four different types. Finally then, we have the results and discussions. The approach for these sections have been modelleged off the state of the arts papers analzyed in the literature review.

B. Algorithm

```
API Call Input:
User selected which type of file they want to wipe metadata from:
    • JPEG
    • PDF
    • docx
    • excel (xlsx)
User uploads files with filename and file type included
All files are added to a list for metadata wiping
        For each file:
                Assess file details:
                        If valid and non-malicious file details:
                                Assess file on Virus Total via API call:
                                        If file clean:
                                            Open the file
                                            Read all metadata and add to list for display on UI
                                            Wipe metadata for that file
                                            Save metadata changes to file
                                        Else:
                                            Reject File and add to error list for UI
                        Else:
```

Reject File and add to error list for UI
Display success and failures of metadata wiping
Downloads files that has been stripped of metadata

Fig.3.1 - Pseudocode displaying how the metadata is wiped

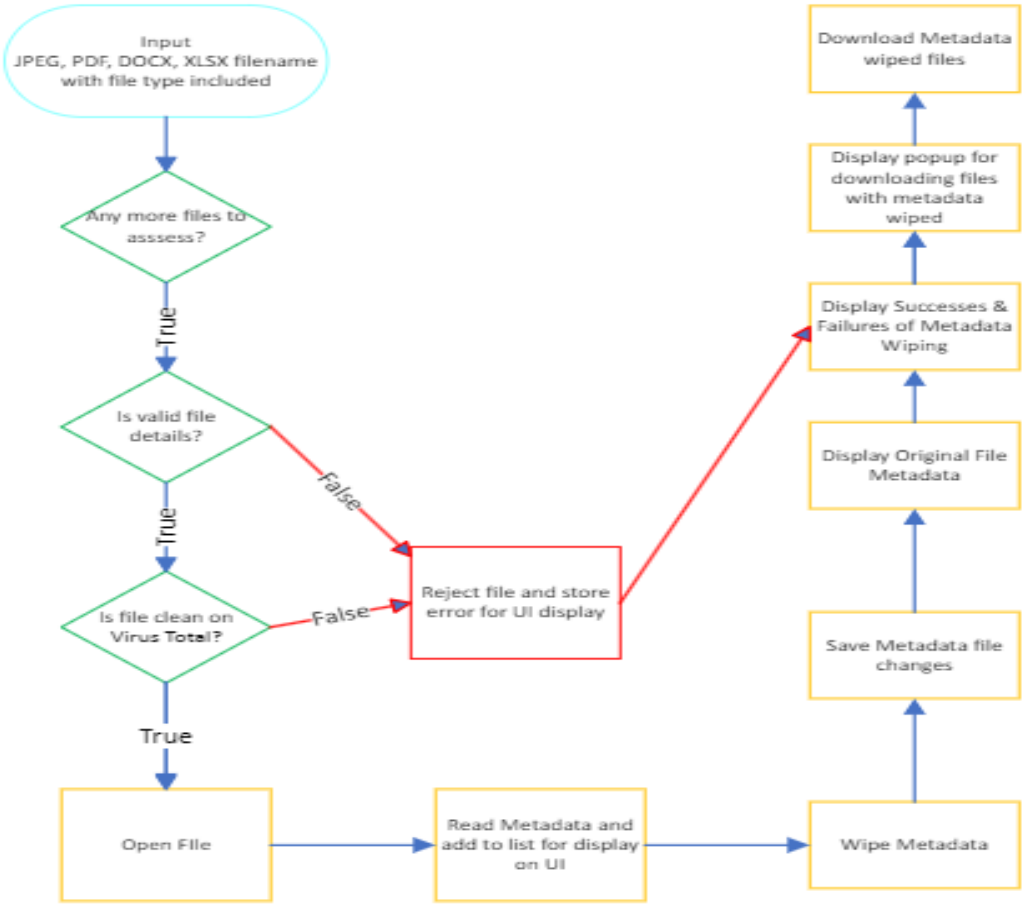

Fig. 3.2 - flowchart of algorithm for metadata wiping

Results

In the next section we look the results obtained from performing the experiments on each of the different file types, JPG, PDF, XLSX and DOCX. It starts with examining the properties of the JPG properties before and after wiping the metadata and then delves into the log output for the presentation of the removal of the GPS co-ordinates from the file. After that we're able to see the before and after effects of the metadata wiping on the remaining three file types of PDF, XLSX and DOCX. Finally then we're able to see the other impacts on the files from the metadata wiping process which includes the size change to the files and the time taken to complete the metadata wiping process.

Metadata Wiping Output - JPG

Table I - Properties metadata wiping is carried out on for JPG files

Properties			
Artist	Model	GPS_Latitude	GPS_Longitude_Ref
Copyright	Software	GPS_Altitude	
Make	GPS_Longitude	GPS_Latitude_Ref	

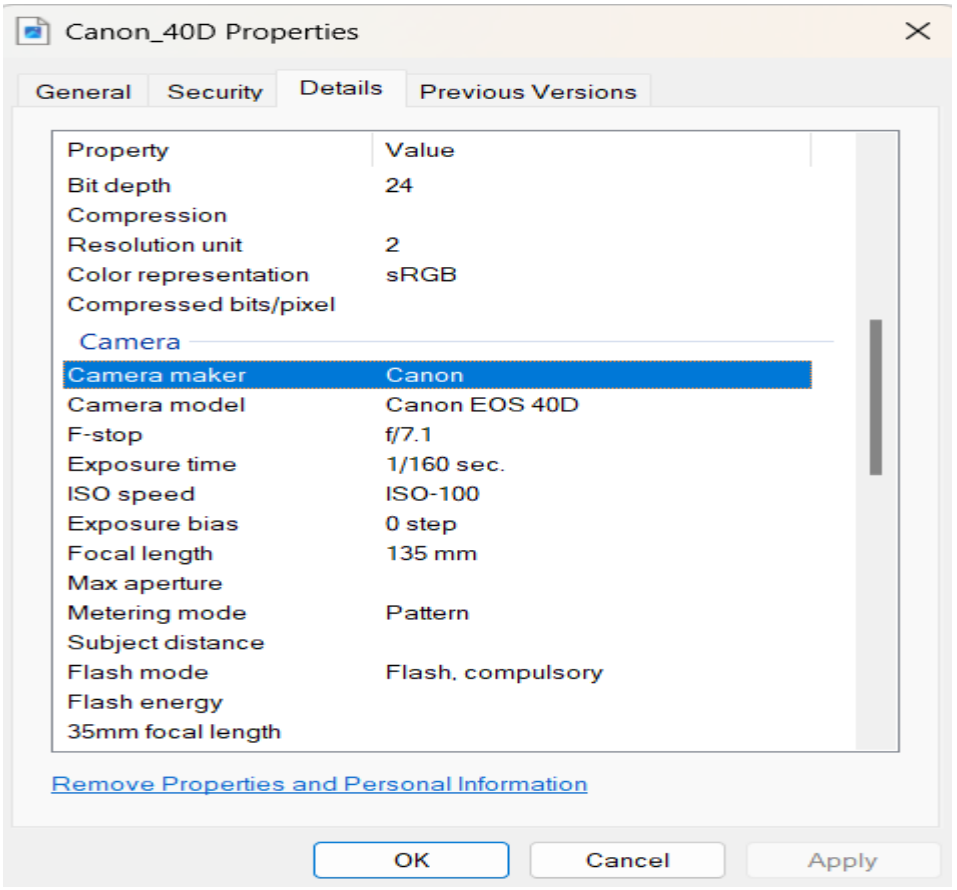

Fig. 3.3 - Image prior to wiping metadata with camera maker and camera model visible

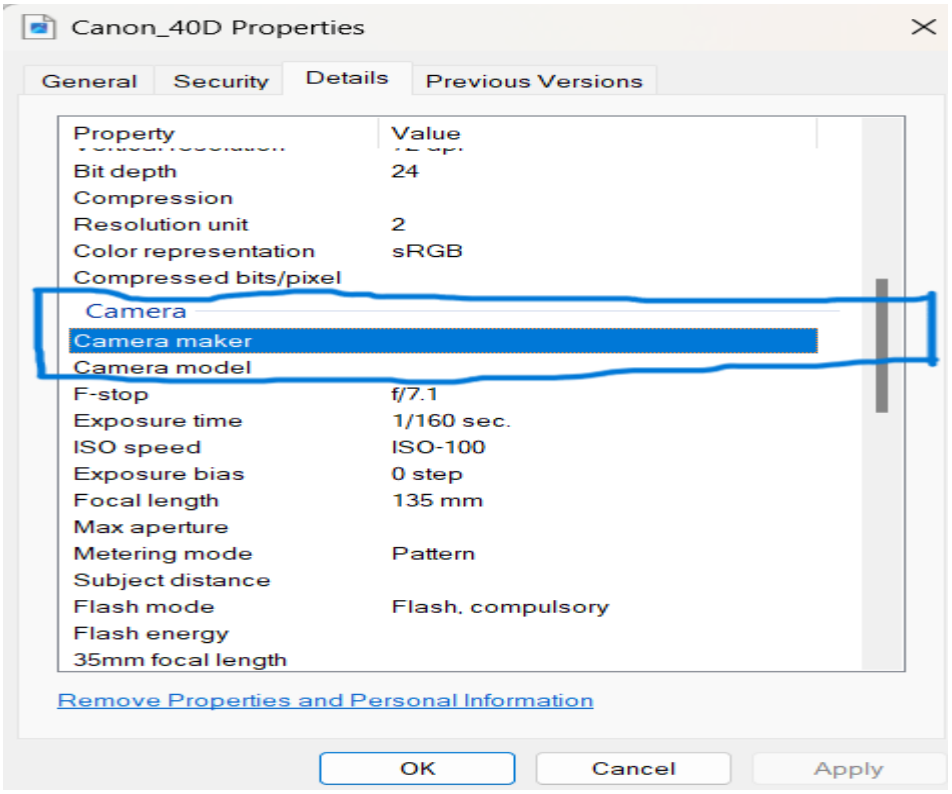

Fig. 3.4 - Camera Maker and Camera Model wiped on a jpeg file

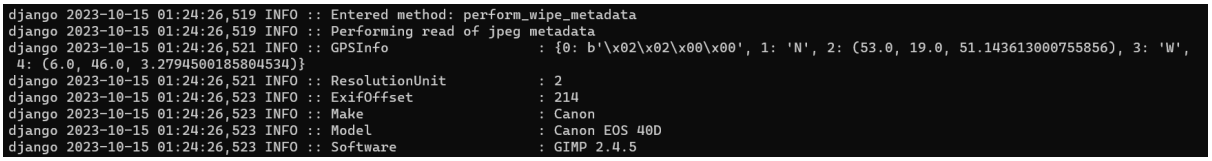

Fig. 3.5 - GPS info of longitude and latitude as hexadecimal and make and model present before wiping metadata

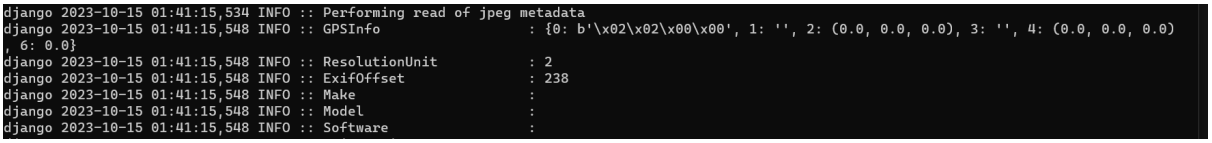

Fig. 3.6 - GPS info of longitude and latitude, make, model & software all wiped after metadata wiping process complete

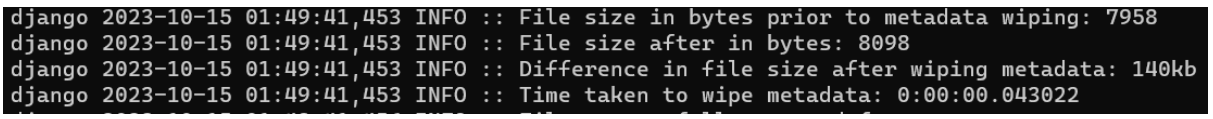

Fig. 3.7 - Time taken and change in file size illustrates miniscule time to complete operation on Jpg file and minor difference in file size after wiping.

Docx

Table II - Properties metadata wiping is carried out on for Docx files

Properties			
Created (date)	Keywords	Modified	Version
Author	Language	Revision	Comments
Category	Last_Modified_By	Subject	
Identifier	Last_Printed	Title	

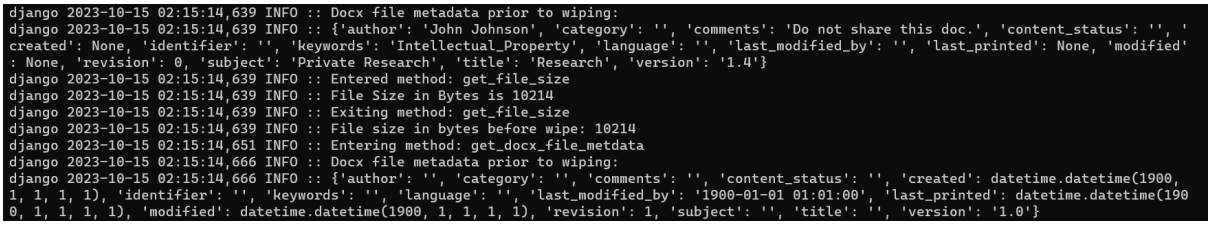

Fig 3.8 - shows a docx file which has metadata originally and then in the bottom of the same image, we can see the metadata has been wiped

Excel

Table III - Properties metadata wiping is carried out on for XLSX files

Properties	
Last_Modified_By	Title
Subject	Category
Keywords	

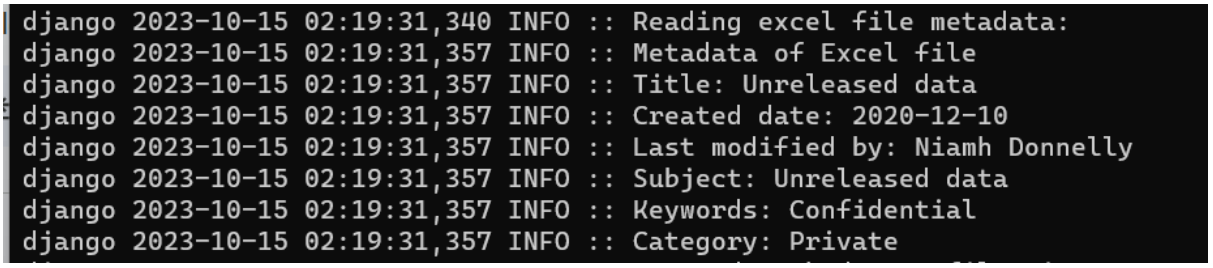

Fig 3.9 - shows the file before the metadata has been wiped.

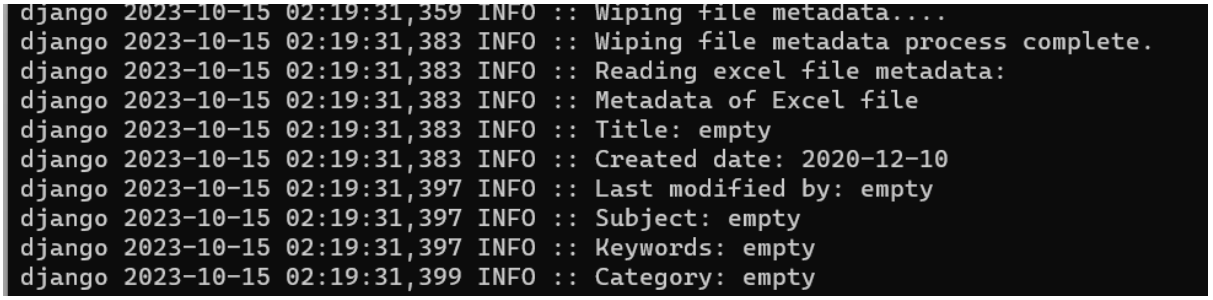

Fig 3.10 - shows the file after the metadata has been wiped. Empty is output when there is nothing present.

PDF

Table IV - Properties metadata wiping is carried out on for PDF files

Properties	
Author	Subject
Title	Keywords
Producer	Creator

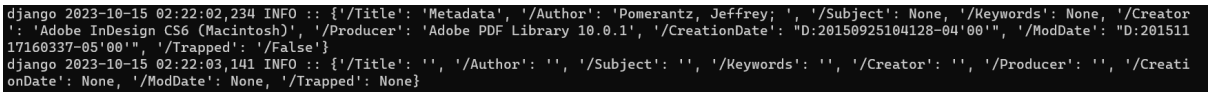

Fig 3.11 - shows the before and after of the file metadata wiping on a pdf file.

Table V- Highlights the time taken to wipe metadata by file type

File Type:	JPG	PDF	XLSX	DOCX
Time to complete:	0.043022s	0.845780s	0.059132s	0.011700s

Table VI - Highlights the file size changes after wiping metadata for each file type

File Type:	JPG	PDF	XLSX	DOCX
File size change:	+140kb	+139377kb	-4657kb	-1210kb

3.1 Discussion

The tool MetadataWiper successfully removed the metadata from each file type on the properties described previously. This tool fills a gap of having a breadth of file types where metadata can be erased in a user friendly manner. It expands on previous research with having some of the most commonly used file types for a spread of different media. This helps users to regain some level of control over their privacy for those who need it and achieves the primary aim of the tool while maintaining the integrity of the file contents. This is a step forward in helping those at an asymmetrical disadvantage when it comes to maintaining their privacy through file metadata.

It puts this layer of protection in place on this facet of privacy which, as shown, can otherwise have a range of consequences for individuals in all walks of life.

While metadata is one important way in which information leaks can occur, there are also other inadvertent leaks that can occur of different varieties within different document types as well. This was identified in the literature review. If this tool were released to production it can also be integrated with other web applications to compound its impact in protecting against information leaks. As such, it could be treated as a third party dependency for performing metadata wiping as part of a sequence of operations. This could be broadly applied with a sequence of other operations aiming to maintain privacy and confidentiality of various file types. APIs could be listed and exposed for this very purpose.

However, on some other fringe performance aspects, the experiments on each file type did show that there was a diverse impact when it came to file size and time taken to complete each operation. The most pronounced impact was on the PDF file which increased in size by 139377kb. This shows an area for an improvement on this particular part of the solution. This is the only metadata wiper service which ended up creating a new file. This seems to have increased the size of the original PDF. The creation of the new file would optimally not happen and this would be explored in any further iteration of a tool like this. However, the integrity of the original file's content was maintained when the new file was created. The XLSX and DOCX solutions in contrast managed to decrease the size of their files with -4657kb and -1210kb respectively. The JPG change was negligible, remaining almost the same with a miniscule rise of 140kb. The time taken to complete the operation was 0.043022s.

As mentioned, the approach for the PDF metadata wiper is inefficient due to the creation of a new file and results in a size increase. Finding a way to alter rather than having to create a new file would be optimal here. This may result in a decrease in the size which occurred on the other file types instead of the increase. Currently, this tool has fixed properties which are deleted each time. An improvement on this would be to allow the users to select which of the properties they wish to delete. This would give more fine grained control to the user if they so required. On top of that, extending to other file types like ,png, .gif, and other very commonly used types could be of benefit as well.

4 Conclusion and Future Work

The objective was to create a privacy tool which can cover multiple of the most common file types and do so in such a way that it makes it accessible to as wide an audience as possible.The integrity of the files ought to be maintained while only altering the metadata. It became abundantly clear from the research conducted that more tools like this one were becoming more and more important across the globe. This is especially the case in countries where surveillance is at its highest. This aim has been successfully achieved with the application capable of removing metadata for the four file formats of .jpg, .pdf, .docx and .xlsx. Having these multiple file formats builds on top of research in this area and hits the gap of covering other very important file types that are ubiquitously used. These file formats are some of the most common and therefore increase the potential reach of the work. With the additional formats, greater control of privacy can be achieved for those who want it in these different file formats. The ability to read a religious text or maintain the freedom of the press ought not to be compromised and individuals sovereignty through privacy protected. The application was produced in an accessible manner with a web application making it available to anyone with a browser and not just to a minority of people who can program. The next step would be to host it and make it production ready.

Increased awareness of the dangers of what might appear innocuous to non-technical users with metadata needs to gain greater public awareness. Only then might there be some chance of change to legal systems to require greater protections be enforced. At some point in the future hopefully metadata will be legally required to be removed and wiped on uploading of documents to any websites unless specifically needed for copyright reasons (for example). Surveillance capitalism ought to be brought to an end to regain individuals' privacy. ISPs ought not to be allowed to record metadata that passes through them and an end. The potential for abuse is too great as highlighted. For now though, tools like this become a necessity for those looking to regain some semblance of protection and privacy on the files they have, upload and transmit online.

References

[1] R. Bhangale, “Securing Image Metadata using Advanced Encryption Standard,”, National College of Ireland., Dublin., 2019.

[2] M. Smith, C. Szongott, B. Henne, G. von Voigt, “Big Data Privacy Issues in Public Social Media,”, Leibniz Universitat Hannover, Hannover, 2013.

[3] B. Henne, C. Szongott, M. Smith, “SnapMe if you can: privacy threats of other peoples; geo-tagged media and what we can do about it” Leibniz Universitat Hannover, Hannover, 2013.

[4] C. Gouert, N. Tsoutsos, “Dirty Metadata: Understanding A Threat to Online Privacy,”, University of Delaware, Delaware, 2022.

[5] B. Toevs, “Processing of Metadata on Multimedia Using ExifTool: A Programming Approach in Python,” In Information and Computer Technology (GOCICT), 2015 Annual Global Online Conference on (pp. 26-30). 2015.

[6] S. Tayeb; A. Week, J. Yee, “Toward metadata removal to preserve privacy of social media users,” CWCC, 2018.

[7] T. Aura, T. A. Kuhn, M. Roe, “Scanning Electronic Documents for Personally Identifiable Information” ACM on (pp. 41-50), 2006.

[8] D. Cole, ‘We Kill People Based on Metadata,’, The New York Review of Books, 2014, nybooks https://www.nybooks.com/online/2014/05/10/we-kill-people-based-metadata/ (accessed October 16th 2023).

[9] R.Pena, M.Rosenberg, “Strava Fitness App Can Reveal Military Sites, Analysts Say.” nytimes.com https://www.nytimes.com/2018/01/29/world/middleeast/strava-heat-map.html#:~:text=Strava's%20online%20exercise%2Dtracking%20map,Afghanistan%20%E2%80%94%20a%20US%20military%20outpost (accessed October 16th 2023).

[10] CVE-2019-3948, nvd.nist.gov https://nvd.nist.gov/vuln/detail/CVE-2019-3948 (accessed October 16th 2023).

[11] S. C. Bennett, J. Cloud, “Coping With Metadata: Ten Key Steps,”Mercer Law Review, Article 2, Volume 61, Mercer Law Review, 2010.

[12] R. M. Smith. “Microsoft Word bytes Tony Blair in the butt, ”, computerbytesman.com, https://www.theguardian.com/technology/blog/2003/jul/03/microsoftword, June 2003.

[13] J.Scott, “Metadata the most potent weapon in his cyberwar,” ICIT, 2017

[14] O.Mundy, “I know where your cat lives”, owenmundy.com https://owenmundy.com/site/i-know-where-your-cat-lives#:~:text=I%20Know%20Where%20Your%20Cat%20Lives%20iknowwhereyourcatlives.com%20is%20a,unknowingly%20uploaded%20in%20their%20metadata (accessed October 16th 2023).

[15] Statistics Times, Top Computer Languages, June 2022, statisticstimes.com https://statisticstimes.com/tech/top-computer-languages.php (accessed October 16th 2023).

[16] J.Ruohonen, A Large-Scale Security-Oriented Static Analysis of Python Packages in PyPI, July 2021, arxiv.org https://arxiv.org/abs/2107.12699 (accessed October 16th 2023).

[17] Veracode, State of Software Security Volume 11 Infosheet, 2023, Online [Available]: https://info.veracode.com/state-of-software-security-volume-11-flaw-frequency-by-language-infosheetresource.html (accessed October 16th 2023).

[18] Veracode, State of Software Security Volume 11, 2023, info.veracode.com https://info.veracode.com/rs/790-ZKW-291/images/Veracode_State_of_Software_Security_2023.pdf (accessed October 16th 2023).

[19] M.Hollander, May 2020, securitycoding.com https://www.securecoding.com/blog/python-security-practices-you-should-maintain/ (accessed October 16th 2023).

[20] Pypi, January 2023, pypi.org https://pypi.org/project/pycryptodome/ (accessed October 16th 2023).

[21] D.Roche, Sqreen, Top 10 Python Security Best Practices, January 2020, blog.sqreen.com https://blog.sqreen.com/top-10-python-security-best-practices/ (accessed October 16th 2023).

[22] M.Hollander, May 2020, securecoding.com https://www.securecoding.com/blog/python-security-practices-you-should-maintain/ (accessed October 16th 2023).

[23] Python Docs, re, docs.python.org https://docs.python.org/3/library/re.html (accessed October 16th 2023).

[24]K.Adair, sydnemcriminallawyers.com.au “Police Using Journalists’ Metadata to Hunt Down Whistleblowers” https://www.sydneycriminallawyers.com.au/blog/police-using-journalists-metadata-to-hunt-down-whistleblowers/ (accessed October 23rd 2023).

[25] E. Eldon. New Facebook Statistics Show Big Increase in Content Sharing, Local Business Pages. http://goo.gl/ebGQH, February 2010.

[26] Facebook. Statistics. 201, facebook.com. http://www.facebook.com/press/info.php?statistics.

[27] E.Snowden, “Permanent Record: A Memoir of a Reluctant Whistleblower”, 2019

[28] M. Furini, V. Tamanini, 2015. Location privacy and public metadata in social media platforms attitudes, behaviors and opinions. Multimedia Tools and Applications, 74(21), 9795-9825

[29] Human Rights Watch, hrw.org, “China: Phone Search Program Tramples Uyghur Rights,”, hrw.org https://www.hrw.org/news/2023/05/04/china-phone-search-program-tramples-uyghur-rights (accessed October 17th 2023).