International Journal of Engineering Sciences & Research Technology

(A Peer Reviewed Online Journal)
Impact Factor: 5.164

IJESRT

RESEARCHERID
THOMSON REUTERS

[Agalya, *et al.*, 9(4): April, 2020]
IC™ Value: 3.00

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

PHISHING SPAM EMAIL ONTOLOGICAL DETECTION

M. Agalya[*1], B. Madhu Varshini[2], S. Dhanushkodi[3] A. Devi[4] & R. Jeevitha[5]
Department of Computer Science and Engineering, Vivekanandha College of Technology for Women, Thiruchengode, Taminadu, India

DOI: 10.5281/zenodo.3778320

ABSTRACT

Targeted Malicious Email (TME) has become more dangerous because it gathers user sensitive information. Beyond spam and phishing designed to trick users into revealing information, TME exploits computer networks and gathers sensitive information. It targets on single users and is designed to appear legitimate and trustworthy. Persistent threat features such as threat actor locale and weaponization tools along with recipient-oriented features such as reputation and role are leveraged with supervised data classification algorithms to demonstrate new techniques for detection of targeted malicious email. We propose a new email filtering technique using random forest classifier and Naïve Bayesian classification. A compromised router detection protocol is developed to identify congestive packet losses. We also develop feature extraction procedure to identify TME specific features. Naïve Bayesian classification is used to classify mails as either TME or trusted mail for user security avoiding frauds. A Naïve Bayesian classifier is a simple probabilistic classifier based on applying Bayesian theorem with strong (Naive) independence assumptions.

KEYWORDS: TME specific feature extraction, NTME congestive packet losses, Random forest classifier.

1. INTRODUCTION

A malicious email message is one which have been deliberately crafted to cause problems on the server or on the client. This could be due to the message containing a virus, or it could be due to the message being crafted in such a way as to take advantage of a weakness in the receiving mail client. GMS provides arrange of checks which may be run against all messages passing through the system to prevent this type of message from entering the server at all. If you do not want to ban the messages entirely it will add a warning to the incoming message. These checks are provided in addition to the standard malicious content checks such as virus scanning and attachment blocking. They are known as Message Quality checks, and inspect the content of each message to ensure it is structurally sound, as well as looking for structures that are typically designed to take advantage of flaws in some of the more popular mail clients. The checks include checking line lengths, checking for the presence of clsid's in mime attachment references, cid's to load html content, suspicious attachments and verifying the integrity of mime encoding.

2. METHODOLOGY

2.1 Waterfall Approach

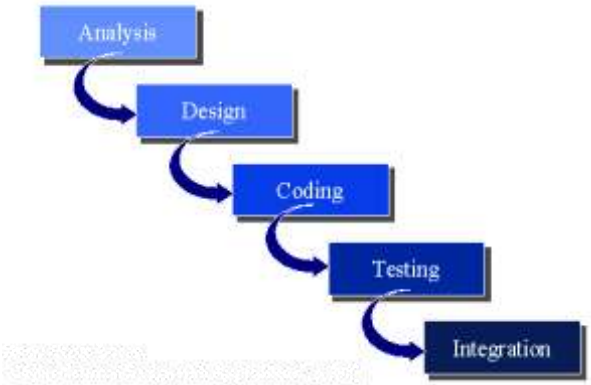
While the Waterfall Model presents a straightforward view of the software life cycle, this view is only appropriate for certain classes of software development. Specifically, the Waterfall Model works well when the software requirements are well understood (e.g., software such as compilers or operating systems) and the nature of the software development involves contractual agreements. The Waterfall Model is a natural fit for contract-based software development since this model is document driven; that is, many of the products such as the requirements specification and the design are documents. These documents then become the basis for the software development contract.

There have been many waterfall variations since the initial model was introduced by Winston Royce in 1970 in a paper entitled: “managing the development of large software systems: concepts and techniques”. Barry Boehm, developer of the spiral model (see below) modified the waterfall model in his book *Software*

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Engineering Economics (Prentice-Hall, 1987). The basic differences in the various models is in the naming and/or order of the phases.

The basic waterfall approach looks like the illustration below. Each phase is done in a specific order with its own entry and exit criteria and provides the maximum in separation of skills, an important factor in government contracting.



While some variations on the waterfall theme allow for iterations back to the previous phase, "In practice most waterfall projects are managed with the assumption that once the phase is completed, the result of that activity is cast in concrete. For example, at the end of the design phase, a design document is delivered. It is expected that this document will not be updated throughout the rest of the development. You cannot climb up a waterfall." (Murray Cantor, Object-oriented project management with UML, John Wiley, 1998)

The waterfall is the easiest of the approaches for a business analyst to understand and work with and it is still, in its various forms, the operational SLC in the majority of US IT shops. The business analyst is directly involved in the requirements definition and/or analysis phases and peripherally involved in the succeeding phases until the end of the testing phase. The business analyst is heavily involved in the last stages of testing when the product is determined to solve the business problem. The solution is defined by the business analyst in the business case and requirements documents. The business analyst is also involved in the integration or transition phase assisting the business community to accept and incorporate the new system and processes.

2.2 V Model

The "V" model (sometimes known as the "U" model) reflects the approach to systems development where in the definition side of the model is linked directly to the confirmation side. It specifies early testing and preparation of testing scenarios and cases before the build stage to simultaneously validate the definitions and prepare for the test stages.

It is the standard for German federal government projects and is considered as much a project management method as a software development approach.

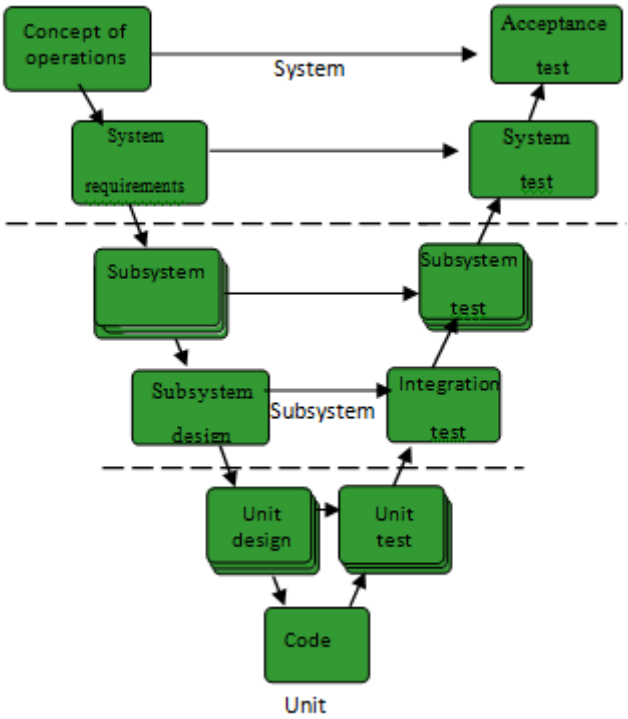
"The V Model, while admittedly obscure, gives equal weight to testing rather than treating it as an afterthought. Initially defined by the late Paul Rook in the late 1980s, the V was included in the U.K.'s National Computing Centre publications in the 1990s with the aim of improving the efficiency and effectiveness of software development. It's accepted in Europe and the U.K. as a superior alternative to the waterfall model; yet in the U.S., the V Model is often mistaken for the waterfall.

"In fact, the V Model emerged in reaction to some waterfall models that showed testing as a single phase following the traditional development phases of requirements analysis, high-level design, detailed design and coding. The waterfall model did considerable damage by supporting the common impression that testing is merely a brief detour after most of the mileage has been gained by mainline development activities. Many

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

managers still believe this, even though testing usually takes up half of the project time." (Goldsmith and Graham, "The Forgotten Phase", Software development, July 2002)

As shown below, the model is the shape of the development cycle (a waterfall wrapped around) and the concept of flow down and across the phases. The V shows the typical sequence of development activities on the left-hand (downhill) side and the corresponding sequence of test execution activities on the right-hand (uphill) side.



The primary contribution the V Model makes is this alignment of testing and specification. This is also an advantage to the business analyst who can use the model and approach to enforce early consideration of later testing. The V Model emphasizes that testing is done throughout the SDLC rather than just at the end of the cycle and reminds the business analyst to prepare the test cases and scenarios in advance while the solution is being defined.

The business analyst's role in the V Model is essentially the same as the waterfall. The business analyst is involved full time in the specification of the business problem and the confirmation and validation that the business problem has been solved which is done at acceptance test. The business analyst is also involved in the requirements phases and advises the system test stage which is typically performed by independent testers – the quality assurance group or someone other than the development team.

The primary business analyst involvement in the system test stage is keeping the requirements updated as changes occur and providing "voice of the customer" to the testers and development team. The rest of the test stages on the right side of the model are done by the development team to ensure they have developed the product correctly. It is the business analyst's job to ensure they have developed the correct product.

3. CONCLUSION

Recall the construction of the NTME1, TME1, and TS1 datasets. At first, we used a 10-fold cross validation as our evaluation method for the joint NTME1–TME1 dataset. Later, we used the joint NTME1–TME1 dataset for training, but instead of doing cross validation, we used the independent TS1 dataset to evaluate the TME filter constructed using the joint NTME1–TME1 dataset.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

4. FUTURE ENHANCEMENTS

For future research, we hope to extend feature extraction to file attachment metadata. Threat actors might inadvertently leave remnants of information such as file paths, time zones, or even author names.10 All these features might associate multiple intrusion attempts into a related campaign. In addition, organizations can track features that characterize the types and amounts of email received by a particular email address. For example, for each recipient, the number of emails and attachments received over a fixed time period might help uncover email that falls outside of his or her normal receiving patterns. For email with hyperlinks, we could develop features to indicate whether the domain of a link has ever been visited before. We could also incorporate information related to domain Creation. Aside from extending email classification features, we could also map features to different threat actors for a multi classification model. As organization and recipient-oriented information evolves, we hope to evolve our techniques accordingly.

REFERENCES

[1] Sikha Bagui Department of Computer Science University of West Florida Pensacola, USA bagui@uwf.edu, "Classifying Phishing Email Using Machine Learning and Deep Learning" in 2019.
[2] Yong Fang , Cheng Zhang, Cheng Huang , Liang Liu, And Yue Yang "Phishing Email Detection Using Improved RCNN Model With Multilevel Vectors and Attention Mechanism in 2019.
[3] Srushti Patil Department of Computer Engineering Sardar Patel Institute of Technology Mumbai, India srushti.patil@spit.ac.in "A Methodical Overview on Phishing Detection along with an Organized Way to Construct an Anti-Phishing Framework" in 2019.
[4] Naghmeh Morad poor School of Computing (SoC) Edinburgh Napier University (ENU)Edinburgh, UK "Employing Machine Learning Techniques for Detection and Classification of Phishing Emails" in 2017.
[5] Rabab Alayham Abbas Helmi Faculty of Information Science &Engineering Management & Science University Shah Alam, Malaysia rabab_alayham@msu.edu.my, "Email Anti-Phishing Detection Application" in 2019.
[6] Asif Karim, Sami Azam, Bharanidharan Shanmugam, Krishnan Kannoorpatti, Mamoun Alazab, "A Comprehensive Survey for Intelligent Spam Email Detection" in 2019. Hongming Che, Qinyun Liu, Lin Zou and HongjiYang Centre for Creative Computing Bath Spa University England," A Content-Based Phishing Email Detection Method" in 2017.
[7] Mohammad Abu Qbeitah_, and Monther Aldwairi_y_College of Technological Innovation Zayed University, Abu Dhabi, UAE 144534 Email: fM80007215, monther.aldwairig@zu.ac.ae in 2017.
[8] Wang Xiujuan Beijing University of Technology, Beijing, China;e-mail: xjwang@bjut.deu.cn, "Detecting Spear-phishing Emails Based on Authentication" in 2019.
[9] Ms. Shweta Dasharath Shirsat IT Department M.L. Dahanukar College of CommerceCity: Mumbai Country: India Email: shwetadshirsat@gmail.com, "Demonstrating Different phishing Attacks Using Fuzzy logic" In 2018.