Proceedings of the 6th International Conference on Communication and Electronics Systems (ICCES-2021)
IEEE Xplore Part Number: CFP21AWO-ART; ISBN: 978-0-7381-1405-7

WebSecAsst - A Machine Learning based Chrome Extension

Neelam Singh Gurjar
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
neelamgurjar@am.students.amrita.edu

Sudheendra S S R
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
sabnavisusudheendra@am.students.amrita.edu

Chejarla Santosh Kumar
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
chejarlasantoshkumar@am.students.amrita.edu

Krishnaveni K.S
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
krishnaveniks@am.amrita.edu

2021 6th International Conference on Communication and Electronics Systems (ICCES) | 978-1-6654-3587-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCES51350.2021.9488953

Abstract— **A browser extension, also known as a plugin or an addon, is a small software application that adds functionality to a web browser. However, security threats are always linked with such software where data can be compromised and ultimately trust is broken. The proposed research work jas developed a security model named WebSecAsst, which is a chrome plugin relying on the Machine Learning model XGBoost and VirusTotal to detect malicious websites visited by the user and to detect whether the files downloaded from the internet are Malicious or Safe. During this detection, the proposed model preserves the privacy of the user's data to a greater extent than the existing commercial chrome extensions.**

Keywords— Machine Learning, Web Security, Chrome Extension, XGBoost, SHAP, Phishing.

I. Introduction

Internet, can be called as the information superhighway, has become inevitable in our lives. However, due to the fact that even a fraud can also be a click away, awareness among users is obligatory.

Phishing is one of the most common cyber-attacks, and according to the FBI [1], there were around 241,342 reported incidents in 2020, making it higher than any other cyberattack in terms of the number of victims. The number of reported phishing attacks in 2020 is more than double the number reported in 2019 which is 108,869. According to Webroot Threat Report 2021 [2], the number of phishing attacks increased 510% from January to February 2020, eBay and Apple were the most targeted sites among them. At the end of 2020, 54% of the phishing sites used HTTPS, which makes it hard for a naive user to detect. According to Microsoft Security Endpoint Threat Report 2019 [3], India ranks 2nd as a victim of drive-by download attacks and 3rd in Asia as a victim of Ransomware attacks which generally spread via the Internet.

The attackers are coming up with new exploits every day making it difficult for a naive user to detect, so having antivirus software in the form of a browser extension can come as a great help. The browser extensions are often light-weight compared to traditional anti-virus software, but most of the popular Chrome security extensions like Avast Online Security [4], AVG Online Security [5], Avira [6], McAfee [7], etc. assists the user about the website safety while browsing the internet through warning but they don't provide assistance about the safety of the downloaded files.

The browser extension we developed not only provides assistance while browsing the internet but also classifies the downloaded files as malicious or safe to open/use along with maintaining user privacy by executing almost everything within the system. While many approaches available for the classification of malicious websites [8] [9] [10] [11] we've chosen our approach to rely on Machine Learning to make if more adaptive to the frequent changes in the attack types.

II. Related Work

The exploration for securing a network has always been one of the top priorities of any organization. Over the years, Machine Learning has become a vital technology in cybersecurity. The Machine Learning algorithms help detect malicious activities and stop attacks. Malicious Web Content Detection using Machine Learning, the authors developed a chrome extension based on random forest model, to predict if the website the user is visiting is malicious or not [12]. There are numerous commercial security chrome extensions available in the market. Some of them are Avast Online Security, Avira Browser Safety, McAfee SECURE. Though Avast promises excellent protection for devices, it was recently uncovered that Avast scrapped data from its users and sold it to third parties which were renowned organizations, data like YouTube views, browsing history, and Google Maps locations [13]. Avira Antivirus application is reviewed to be effective with a solid firewall, helps prevent spams. However, all this comes with quite a price. McAfee scores top marks when it comes to excellent protection, proving to give security against malware. One of the biggest imperfections of this software is that it tends to make the system a little slow, and might lead to boot loops.

III. Architecture of WebSecAsst

There are two functionalities of the extension, 1. Analyze the current website 2. Analyze the downloaded files.

On opening the extension, the first functionality will be triggered automatically. The extension will send the URL of the website in which the extension is opened, to "ENDPOINT 1" as shown in Fig. 1, which is running in the local server of the user. This will trigger a Python script for analyzing the website. The Python script uses a pre-built XGBoost model to analyze the website. Once the classification is done, the result will be sent back to the ENDPOINT 1 which will return the same to the extension.

The second functionality is triggered manually when the user clicks on the "Verify" button on the right side of the downloaded file. On clicking the button, the "path" of the file will be sent to "ENDPOINT 2" as shown in Fig. 1, running in the local server of the user, which will trigger another Python script that would classify the downloaded file as MALICIOUS or SAFE by sending the MD5 hash of the file to ViruaTotal API [14] and the result is returned to the ENDPOINT 2 which will return the same to the extension.

The script [15] we have developed for setting up the extension would automatically setup a local Apache2 server [16], install PHP [17], Python3 [18], and set up a Python virtual environment [19] dedicated for the extension and would create a folder WebSecAsst at location /var/www/html in the users system that would contain all the scripts required for the proper functioning of WebSecAsst.

A. Advantages of the Framework

Unlike most of the extensions which are present in the Chrome Store, which will send the Website URLs the user is visiting to their servers in order to analyze the website, our extension will perform the analysis in the local system itself. The only information that is sent out of the user's system is the hash of the downloaded file, which is computationally not feasible to reverse, thus preserving the privacy of the user to a better extent.

IV. Working of ENDPOINT 1

Once the ENDPOINT 1 [20] receives a request from the extension, the ENDPOINT 1 will trigger the python script [21] that would classify the website the user is visiting as MALICIOUS or SAFE, the script will parse the entire HTML and Javascript code of the website using packages requests and BeautifulSoup, pulls and analyses the CA Certificate information of the website using ssl, OpenSSL and socket packages of python. Upon analyzing the data the script would create an input array for the XGBoost model we've built, gets the prediction from the model and returns the classification result to the ENDPOINT 1 which would return the same to the extension to display the classification to the user.

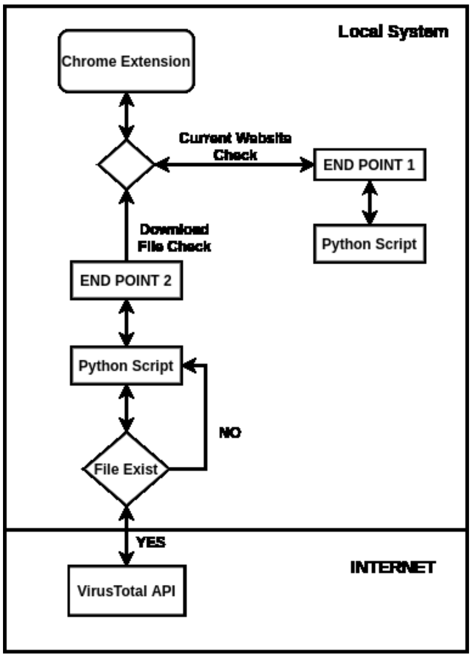


Fig. 1. Architecture of WebSecAsst

V. Working of ENDPOINT 2

Once the ENDPOINT 2 [22] is triggered by the extension, the ENDPOINT 2 will trigger a python script [23] to classify the downloaded file as MALICIOUS or SAFE. The script will first checks for the existence of the file, if the file is not present at the path received by the script, the script will return the message "File Deleted/Moved to other location" to the ENDPOINT 2 which will return the same to the extension to display for the user. I the file exists in the path the script will calculate the MD5 hash of the file and will send the hash to VIRUSTOTAL API which will return a JSON response containing information about the file derived from various security vendors, upon analyzing the JSON response even if one of the security vendor suggests that the file is MALICIOUS the python script will send the response as MALICIOUS back to the ENDPOINT 2 else if all the security vendors suggests that the file is SAFE the python script will send the response as SAFE back to the ENDOINT 2, which will send back the received response back to the extension to display the classification to the user.

VI. Machine learning

A. Dataset

The dataset [24] we chose for building the model has 30 attributes and 2456 instances, out of which we used 24 attributes to build our model, we've ignored the features "port", "on mouseover", "popUpWidnow", "Page Rank", "Links pointing to page", "Statistical report" as they were either infeasible to calculate or inaccessible (requiring data from the servers which can't be accessed). Each Feature in the dataset has one of the three possible values -1, 0, 1. '-1' implies that the feature is considering the website as "Phishing", '0' implies that the feature is considering the website as "Suspicious", and '1' implies that the feature is considering the website as "Legitimate".

B. Models

We have built and compared different machine learning models. The final model we chose was built using the XGBoost algorithm, giving an accuracy of 96.69%. Comparison between different algorithms can be seen in the Table 1 & Table 2.

TABLE I. ACCURACY COMPARISON

Algorithm	Accuracy
XGBoost	96.698
Random Forest	95.838
KNN (n=3)	94.029
KNN (n=5)	93.668
KNN (n=7)	92.220

TABLE II. COMPARISON OF TN,FP,FN,TP

Algorithm	TN	FP	FN	TP
XGBoost	934	39	34	1204
Random Forest	938	61	31	1181
KNN (n=3)	917	82	50	1162
KNN (n=5)	912	87	53	1159
KNN (n=7)	887	113	60	1152

C. Model Explanation

We have performed a thorough analysis of the model we've built. There are various approaches of Explainable Artificial Intelligence (XAI), which can be used to understand a Machine Learning model, We've chosen SHAP [25] [26] [27] [28] [29] for understanding our model. SHAP - SHapley Additive exPlanations, explains the output of a Machine learning model by calculating Shapley values for each feature of the model using a Game Theoretic approach. The higher the Shapley value, the higher the impact of the feature on the model's output. Fig 2 shows a bar chart with the average impact of each feature on the model's output.

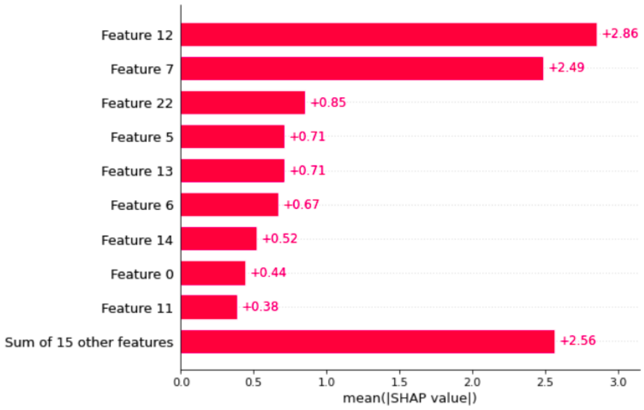


Fig. 2. Mean of SHAP values plot

The top 5 features which have the most impact on our model's output are:

- URL_of_Anchor: All the anchor tags, which are represented as <a> in HTML of the web page are analyzed and if the percentage of unsafe anchor tags are more than 67%, '-1' is assigned to the feature, if the unsafe anchor tags percentage is between 31% and 67%, '1' is assigned, else '0' is assigned to the feature. An anchor tag is considered unsafe if it is pointing to a website of another domain, and/or the tag is having #, #content, #skip, #JavaScript ::void(0) as href value.

- SSLfinal_State: Most of the malicious websites don't have an SSL certificate or might have a certificate issued by an untrusted Certificate Authority or the SSL Certificate will be less than a year old in general. Value '1' is assigned to the feature if the website uses HTTPS and the SSL Certificate is issued by a Certificate Authority which is trusted and the age of the certificate is more than one year. If the website is using HTTPS and the Certificate Authority is not trusted, the feature is assigned '0', else '-1' is assigned to the feature.

- Shortining_Service: Use of an URL shortening service to hide the actual URL is one of the common practices of the attackers. If the website is using this service, the feature is assigned '-1' else '1'.

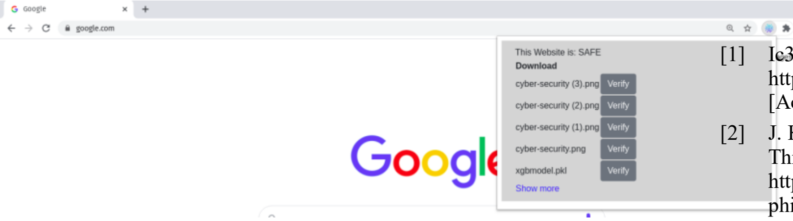
- web_traffic: Alexa[30] database has records of most popular websites based on the web traffic. Since most of the phishing websites live for a short period and the traffic on these websites will be relatively less, the Alexa database might not have a record. This feature is assigned '1' if the Website rank from the Alexa database is less than 1,00,000, '0' if the Website rank is greater than 1,00,000 , '-1' if there's no record for the website.

- Prefix_Suffix: Most of the phishing websites have a '-' in their domain name, attackers place '-' between the characters to make the domain look like an actual one. This feature is assigned '-1' if there is a '-' character in the domain of the website else '1'.

VII. Result

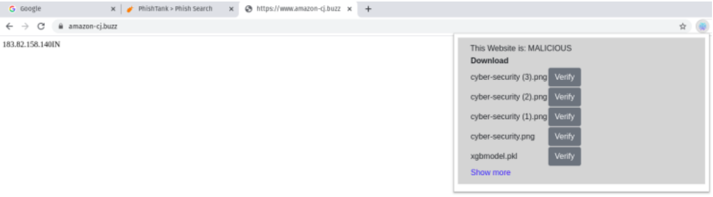
The code and setup process of WebSecAsst can be found on Github[31]. The interface and working of the WebSecAsst can be seen in the Figures, Fig. 3., Fig. 4, Fig. 5., Fig. 6.

- Verification of https://google.com, which is not a malicious website can be seen in Fig. 3., the prediction made by the model is also “SAFE”.
- Verification of https://www.amazon-cj.buzz/ and http://69.49.229.16/public/banks/tangerine/ which are malicious websites can be seen in Fig. 4. and Fig. 5., the predictions made by the model are also “MALICIOUS”. The links of the malicious websites are obtained from phishtank.com[32].
- Verification of downloaded files can be seen in Fig. 6., Where BOMBERMANIA.EXE.ZIP is a malicious file downloaded from tekdefense.com[33] and the response we got from VirusTotal is also malicious. The second file is intentionally deleted to show the functionality of the extension - when a file is deleted or moved from the original download folder, and file “cyber-security.png” is an image downloaded from google images which is a safe file and the response from VirusTotal is also “SAFE”.



This Website is: SAFE
Download
cyber-security (3).png Verify
cyber-security (2).png Verify
cyber-security (1).png Verify
cyber-security.png Verify
xgbmodel.pkl Verify
Show more

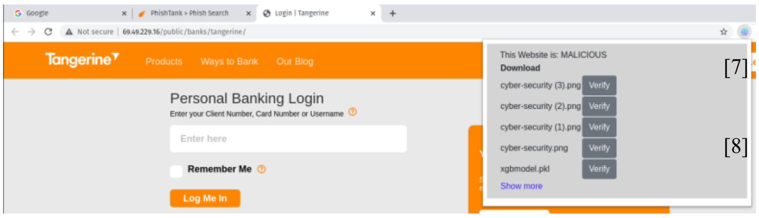
Fig. 3. SAFE



183.82.158.140IN

This Website is: MALICIOUS
Download
cyber-security (3).png Verify
cyber-security (2).png Verify
cyber-security (1).png Verify
cyber-security.png Verify
xgbmodel.pkl Verify
Show more

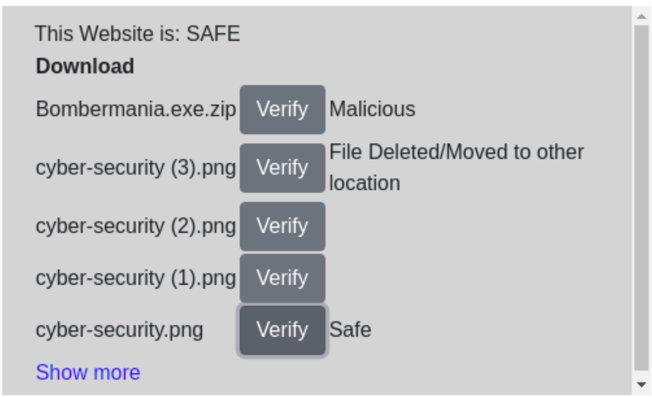
Fig. 4. Malicious



Personal Banking Login
Enter your Client Number, Card Number or Username
Enter here
□ Remember Me

This Website is: MALICIOUS
Download
cyber-security (3).png Verify
cyber-security (2).png Verify
cyber-security (1).png Verify
cyber-security.png Verify
xgbmodel.pkl Verify
Show more

Fig. 5. Malicious



This Website is: SAFE
Download
Bombermania.exe.zip Verify Malicious
cyber-security (3).png Verify File Deleted/Moved to other location
cyber-security (2).png Verify
cyber-security (1).png Verify
cyber-security.png Verify Safe
Show more

Fig. 6. Downloads Verification

VIII. Conclusion & Further Work

As the number of cyber-attacks increase and the approaches of the attackers to exploit the user’s system are changing every day, having a tool that can adapt these changes like WebSecAsst would come of great help to the users. As of now the WebSecAsst only works on Debian-based systems, extending the support to other operating systems will add a great value to the project.

References

[1] Ic3.gov, 2021. [Online]. Available: https://www.ic3.gov/Media/PDF/AnnualReport/2020_IC3Report.pdf. [Accessed: 04- May- 2021].

[2] J. Kurtz, “Ransomware, BEC and Phishing Still Top Concerns, per 2021 Threat Report | Webroot”, Webroot Blog, 2021. [Online]. Available: https://www.webroot.com/blog/2021/04/21/ransomware-bec-and-phishing-still-top-concerns-per-2021-threat-report/. [Accessed: 03- May- 2021].

[3] M. India, “Malware, ransomware and drive-by download attacks pose biggest cyberthreat challenge in India: Microsoft Security Endpoint Threat Report 2019 - Microsoft Stories India”, Microsoft Stories India, 2021. [Online]. Available: https://news.microsoft.com/en-in/microsoft-security-endpoint-threat-report-2019-india/. [Accessed: 03- May- 2021].

[4] “Avast Online Security”, Chrome.google.com, 2021. [Online]. Available: https://chrome.google.com/webstore/detail/avast-online-security/gomekmidlodglbbmalcneegieacbdmki. [Accessed: 03- May- 2021].

[5] “AVG Online Security”, Chrome.google.com, 2021. [Online]. Available: https://chrome.google.com/webstore/detail/avg-online-security/nbmoafcmbajniiapeidgficgifbfmjfo. [Accessed: 03- May- 2021].

[6] “Avira Browser Safety”, Chrome.google.com, 2021. [Online]. Available: https://chrome.google.com/webstore/detail/avira-browser-safety/flliilndjeohchalpbbcdekjklbdgfkk. [Accessed: 03- May- 2021].

[7] “McAfee SECURE”, Chrome.google.com, 2021. [Online]. Available: https://chrome.google.com/webstore/detail/mcafee-secure/lkdiimaiohgpacfbgedcipmgigppaofn. [Accessed: 03- May- 2021].

[8] A. H.M., Dr. Tripty Singh, G., A., and Joseph, G., “Web Security: A prototype Tool for Detecting Web Application Vulnerability”, International Conference on Emerging Trends in Engineering, Business and Disaster Management(ICBDM 2015). Noorul Islam University, Kumaracoil, Tamilnad, 2015

[9] .Seshagiri, Prabhu & Vazhayil, Anu & Sriram, Padmamala. (2016). AMA: Static Code Analysis of Web Page for the Detection of Malicious

Scripts. Procedia Computer Science. 93. 768-773. 10.1016/j.procs.2016.07.291.

[10] .A. Desai, J. Jatakia, R. Naik and N. Raul, “Malicious web content detection using machine leaning,” 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 1432-1436, doi: 10.1109/RTEICT.2017.8256834.

[11] .Varsha, Ravichandra Mouli & Kp, Jevitha. (2016). Web Services Attacks and Security- A Systematic Literature Review. Procedia Computer Science. 93. 870-877. 10.1016/j.procs.2016.07.265.

[12] "Malicious web content detection using machine leaning", Ieeexplore.ieee.org, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/8256834. [Accessed: 24- May- 2021].

[13] “5 top machine learning use cases for security”, Maureen Data Systems, 2021. [Online]. Available: https://www.mdsny.com/5-top-machine-learning-use-cases-for-security/. [Accessed: 24- May- 2021].

[14] “VirusTotal”, Virustotal.com, 2021. [Online]. Available: https://www.virustotal.com/gui/. [Accessed: 03- May- 2021].

[15] ”chsantoshkumar211/WebSecAsst”, GitHub, 2021. [Online]. Available: https://github.com/chsantoshkumar211/WebSecAsst/blob/master/setup/setup.sh.

[16] D. Group, ”Welcome! - The Apache HTTP Server Project”, Httpd.apache.org, 2021. [Online]. Available: https://httpd.apache.org/.

[17] ”PHP: Hypertext Preprocessor”, Php.net, 2021. [Online]. Available: https://www.php.net/.

[18] ”Welcome to Python.org”, Python.org, 2021. [Online]. Available: https://www.python.org/.

[19] ”12. Virtual Environments and Packages — Python 3.9.5 documentation”, Docs.python.org, 2021. [Online]. Available: https://docs.python.org/3/tutorial/venv.html.

[20] ”chsantoshkumar211/WebSecAsst”, GitHub, 2021. [Online]. Available: https://github.com/chsantoshkumar211/WebSecAsst/blob/master/setup/WebSecAsst/phis.php.

[21] ”chsantoshkumar211/WebSecAsst”, GitHub, 2021. [Online]. Available: https://github.com/chsantoshkumar211/WebSecAsst/blob/master/setup/WebSecAsst/phis.py.

[22] ”chsantoshkumar211/WebSecAsst”, GitHub, 2021. [Online]. Available: https://github.com/chsantoshkumar211/WebSecAsst/blob/master/setup/WebSecAsst/virus.php.

[23] ”chsantoshkumar211/WebSecAsst”, GitHub, 2021. [Online]. Available: https://github.com/chsantoshkumar211/WebSecAsst/blob/master/setup/WebSecAsst/virus.py.

[24] ”UCI Machine Learning Repository: Phishing Websites Data Set”, Archive.ics.uci.edu, 2021. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/phishing+websites. [Accessed: 03- May- 2021].

[25] S. Lundberg and S. Lee, ”A Unified Approach to Interpreting Model Predictions”, Proceedings.neurips.cc, 2021. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html. [Accessed: 03- May- 2021].

[26] Lundberg, S.M., Erion, G., Chen, H. et al. From local explanations to global understanding with explainable AI for trees. Nat Mach Intell 2, 56–67 (2020). https://doi.org/10.1038/s42256-019-0138-9

[27] Lundberg, S.M., Nair, B., Vavilala, M.S. et al. Explainable machinelearning predictions for the prevention of hypoxaemia during surgery. Nat Biomed Eng 2, 749–760 (2018). https://doi.org/10.1038/s41551- 018-0304-0

[28] ”slundberg/shap”, GitHub, 2021. [Online]. Available: https://github.com/slundberg/shap. [Accessed: 03- May- 2021].

[29] C. S. Kumar, M. N. S. Choudary, V. B. Bommineni, G. Tarun and T. Anjali, ”Dimensionality Reduction based on SHAP Analysis: A Simple and Trustworthy Approach,” 2020 International Conference on Communication and Signal Processing (ICCSP), 2020, pp. 558-560, doi: 10.1109/ICCSP48568.2020.9182109.

[30] ”Alexa - Top sites”, Alexa.com, 2021. [Online]. Available: https://www.alexa.com/topsites. [Accessed: 03- May- 2021].

[31] ”chsantoshkumar211/WebSecAsst”, GitHub, 2021. [Online]. Available: https://github.com/chsantoshkumar211/WebSecAsst. [Accessed: 03- May- 2021].

[32] 2021. [Online]. Available: https://www.phishtank.com/. [Accessed: 03- May- 2021].

[33] ”TekDefense - News”, Tekdefense.com, 2021. [Online]. Available: http://www.tekdefense.com/. [Accessed: 03- May- 2021]