JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

The Art of Crafting CTF Challenges: Insights and Lessons Learned

Jasjyot Singh Saini[1] **Deep Parasiya[2]** **Chirag Prajapati[3]** **Dr. Nilakshi Jain[4]** **Pranali Pawar[5]**
Engineering Student[1] Engineering Student[2] Engineering Student[3] Head Of Department[4] Co-Guide[5]
Department of Cyber Security[1,2,3,4,5]
Shah & Anchor Kutchhi Eng. College, Mumbai, India[1,2,3,4,5]

Abstract: This Cybersecurity professionals are needed more and more, yet there aren't enough qualified candidates to fill the position. We may use CTF, or capture the flag event, which turns to learn about vulnerabilities into a fun game, to educate young people about the rapidly evolving field of cyber security. However, how do we create challenges for our CTF? This study addresses the issue. We delve deeply and go through each and every step necessary to create an engaging CTF challenge. To make it more illuminating, we revealed the development process of five of the CTFs we created and provided a list of helpful resources

Index Terms **- Cyber Security, Hacking, Capture the Flag, CTF, Cyber Forensics, Networking, Steganography, Guide.**

I. INTRODUCTION

About

CTF or Capture The Flag Challenge is a type of game or competition where players must complete a variety of challenges. These tasks simulate real-world vulnerabilities that can be found on the internet and in other forms of digital media. Participants can learn about the vulnerabilities being shown through these problems, as well as how to take advantage of them. The best part is that you learn by practically exploiting the vulnerability to reach the flag. Logic-based strings of characters called flags, such as "ITCTF{JETIR_SUBMITTED}" signal the conclusion of a challenge and serve as proof that the problem has been solved by the player. The competitor with the most flags at the end of the course wins the event. Depending on the organizers, CTF challenges can last up to 48 hours or 3 hours and can be team-based. In this paper, we'll talk about how to create challenges for a CTF and how we accomplished it

Motivation behind the project

As technology advances, there is an increasing need for qualified cyber security professionals to safeguard infrastructure and important data. These CTF competitions serve as an excellent introduction to the field of cyber security for young, inquisitive minds that may go on to become entrepreneurs in this field. For those who already know how to, this will be a terrific opportunity to practice their abilities and learn new approaches to problem-solving. And as for the CTF creators, we learned a great deal about the challenges industry, and to top it all off, the idea that one day someone might be inspired by the challenges we made to work in this field is just incredible.

II. METHODOLOGY

Challenge Planning

We must first select how many domains we will have and how many problems will be present in each domain before we can start building any challenges. In total, there are 6 domains:

- Web Based-These tasks simulate the flaws you could encounter in web applications, like SQL injection, etc.
- Forensics: These challenges imitate how real-world cyber investigators evaluate data. Example: memory dumps or .pcap files
- Cryptography: Cryptography focuses on several established and novel techniques for data encryption and decryption.
- Reverse Engineering: These challenges in reverse engineering concentrate on malware analysis and decompiling different unsafe applications. They typically involve low-level programming languages like assembly.
- OSINT: These challenges essentially evaluate your ability to obtain information and navigate Google.
- Miscellaneous: Everything else falls under this category, thus understanding the fundamentals of cyber security can help with them.

Choose the domain and number of challenges based on the total number of developers and their areas of expertise. If you choose too many tasks, it will be impossible to complete them all on time, and if you choose too few, you won't learn anything new. For our CTF, we were a team of 3 and chose two domains to focus on: Web-Based & Forensic

Memory Forensics CTF Challenge

a) Introduction: This memory dump-based challenge is intended to test participants' understanding of how to examine memory dumps using the Volatility Framework. Volatility is an open source, python-based implementation for extracting digital artifacts from volatile memory (RAM) samples

b) Challenge Design: To create this challenge, we must create a memory dump file (.vmem) that participants can examine and locate the flag in. We'll need a virtual machine to create a memory dump, and for the purpose of this project, VMware worked best because it makes it simple to extract memory files. Because it is very simple to install and its ISO image is quite simple to find online, Windows 7 will be the operating system we use. Once Windows 7 is operating on VMware, the setup is finished. Now, a typical way to hide the flag is to save it in the clipboard, but for our challenge, we used the echo command to print the flag in cmd. You can use any location you like, just make sure it can be extracted. Once the flag has been hidden, suspend the virtual machine from VMware and browse to its source files, where you will discover a ".vmem" file. Your memory dump file for the challenge is that one.

c) Solution Strategy: Understanding how to use Volatility Framework is essential in order to complete the task. The first step for participants is to determine which kind of OS the memory dump file was running on. "volatility -f Mrmango-d6d99913.vmem imageinfo" can be used to accomplish this. You can use that to find out that it runs Windows 7. Participants must now search in cmd for the flag that is hidden. "volatility -f Mrmango-d6d99913.vmem -- profile=Win7SP1x64 cmdscan" is the command to read the cmd history. This will make the flag visible, and the Challenge will be solved. Fig. 1. Output given by volatility after running CMDSCAN

Fig.1 output given by volatility after running CMDSCAN

d) Conclusion: In conclusion, the participants were able to put their memory analysis skills to the test and looked for the flag in areas where there might be buried important information in realworld circumstances. The difficulty of the challenge can be increased by including a malware analysis component.

ROT47 CTF Challenge

a) Introduction: The purpose of the cryptography CTF challenge was to test players' understanding of and proficiency in ROT47, hexadecimal, and binary encoding. The challenge was to use these methods to reveal a flag that had been encoded.

b) Challenge Design: The flag was initially transformed into binary, a base-2 numbering system composed of 0s and 1s. After that, the binary data was converted into hexadecimal, a base-16 numbering system with the digits 0 through 9 and A–F. Finally, ROT47 was used to encrypt the hexadecimal data. ROT47 is a straightforward ASCII character substitution cypher that swaps out each character for the character that is 47 positions distant from it in the ASCII table. "I played with 1s and 0s since I was in 6th and my roll no. was 47," was the challenge's hint. This clue implies that the problem somehow involved binary and the number 47.

c) Solution Strategy: Participants have to reverse the encoding procedure in order to complete the challenge. To get the original hexadecimal value, they first had to decode the ROT47-encoded hexadecimal data. The hexadecimal data had to then be changed back into binary. In order to retrieve the flag, they had to transform the binary data into ASCII characters. The hint that was given helped to solve the problem as well. When playing with 1s and 0s is mentioned, it is likely that the challenge used binary, and when the roll number is mentioned, it is likely that the flag was encoded using ROT47.

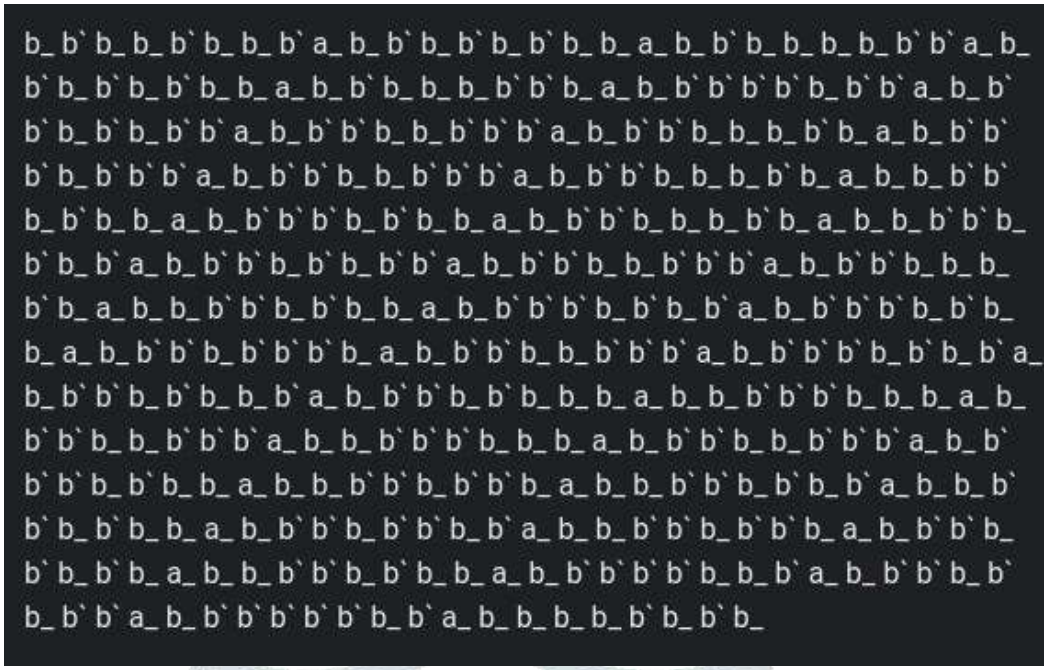

Fig.2 Hex data encoded in rot47

d) Conclusion: The cryptography CTF challenge was a success overall, as it successfully assessed participants' understanding of and expertise in ROT47, hexadecimal, and binary encoding. The assignment became more intriguing and challenging as a result of the hint's use. Additional encoding methods or other types of encryptions might be used into challenge iterations in the future to increase the difficulty level for competitors

Magic Bytes CTF challenge

a) Introduction: The CTF challenge makes use of the concept of magic bytes, which are the first four to five hexadecimal bytes of a file that an operating system scans to determine what kind of file it is. These hexadecimal bytes can be changed to easily deceive the OS.

b) Challenge Design: First, we make a ".jpg" file, which is an image file. It will have the flag written on it. Now we will use the hex editor cli application, which can be installed from the apt library. Using the hex editor, we will convert the first 4 hex bytes of the.jpg file to hex bytes of a PDF or any other file type you want. PDFs hex bytes are short, which makes them an ideal candidate. Once we have edited and saved, it will look like a pdf, but when opened using a pdf viewer, it won't work. This PDF is our challenge manuscript.

c) Solution Strategy: The fact that the challenge is called "Magic Bytes" will offer them a hint as to where to begin and what to search for. Any hex editor may be used by participants to view the file's contents. "2550 4446 2d10 4a46 4946 0001 0102 001c %PDF-.JFIF..." begins the first line. Hexadecimal is on the left side of this image, and its text interpretation is on the right. Take note of the JFIF; these often do not exist in PDF files. A quick search on Google reveals that it is a JPG File. When the file is fixed and opened with an image viewer, the flag will be visible. Editing them and making the initial bits identical to jpg files will correct the file. Fig. 3. First Hexadecimal (Magic byte) of our file Fig. 4. Text interpretation of above Hexadecimals

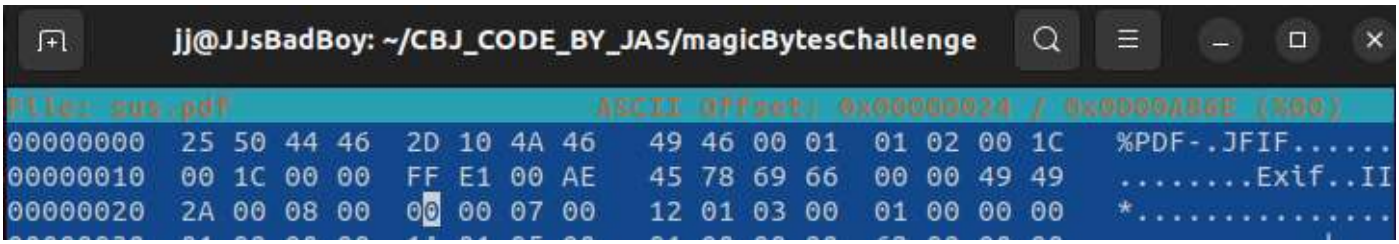

Fig.3 hex bytes of the file

d) Conclusion: Participants in this challenge are taught how the file identification system functions and how it can be manipulated to trick the operating system and users

Web-Based CTF Challenge

a) Introduction: The purpose of the web-based CTF challenge was to put participants' knowledge of system administration, network scanning, and web creation to the test. The task required scanning an IP address, finding a hidden SSH username and password, utilizing SSH to log into an Ubuntu machine, finding two hidden flags and figuring out a background process that would disclose the third flag.

b) Challenge Design: The Ubuntu base image and Docker were used in the challenge's design. Nmap was used to scan the IP address given to the participants, and it showed that ports 80 and 22 were open. A typical website was hosted on port 80 using the Apache2 server, and the SSH service was active on port 22. Two files were hosted on the website; the first, /asphalt.html, had a standard image while the second, /asphalt.html, included the SSH username base64-encoded in the HTML comments. The first flag and the base64-encoded SSH password were both located in the second file, robots.txt. Participants could access the Ubuntu machine's home directory by logging in once they got the SSH login information. In the home directory, the second flag was concealed. The last flag was concealed in a background-running cronjob. A request with the last flag in the URL would be sent to the localhost by the cronjob.

c) Solution Strategy: Participants had to use Nmap to scan the given IP address in order to find that ports 80 and 22 were open before they could complete the challenge. The next step was to access the website running on port 80 and find the base64-encoded hidden SSH username in the comments of the /asphalt.html file. After that, participants had to decode the SSH password's base64 encoding and the first flag given in the robots.txt file. With this knowledge, users may use SSH to enter into the Ubuntu computer and find the second hidden flag in the home directory. The last task required participants to pinpoint the background process that was making a request to localhost that included the last flag in the URL. This needed an understanding of how cronjobs operate as well as the ability to recognize and decode the request's final flag.

```
<!DOCTYPE html>
<html lang="en">
    <head><title>
        CTF web page
    </title>
    <style>
        img {
            width: : 1500spx;
            float:left;
            margin-right: 50px;
        }
        .username {
            font-weight: bold;
        }
    </style>
</head>
<body>
    <h1 style="text-align:center"> Welcome to Asphalt Games</h1>
    <img src="https://i.ytimg.com/vi/4UwkS8ODFv0/maxresdefault.jpg" width="100%" height="100%"  ">

    <!-- ZGVlcA== -->

</body>
</html>
```

Fig. 4. Flag hiding in HTML Comments

d) Conclusion: The web-based CTF challenge, in its whole, was a difficult, multi-layered task that successfully tested participants' knowledge and proficiency in network scanning, web creation, and system management. The challenge was made more difficult and realistic by the use of Docker and the Ubuntu base image. It was particularly difficult and required considerable understanding of cronjobs to complete the final phase, which involved determining the background process and decoding the final flag. In order to push players even further, future challenges can include new complexity layers or incorporate different cybersecurity ideas

Steganography CTF challenge

a) Introduction: Steganography CTF challenges are a type of cybersecurity competition that involve the use of steganography techniques to hide or discover hidden messages within various types of media, such as images, audio, and video files. These challenges are designed to test the skills of participants in identifying and manipulating hidden messages that are embedded in various digital media

b) Challenge Design: Locate an audio file that will serve as the foundation for the task. Any kind of audio file, including a song or sound effect, could be included. Use steganography to encrypt the audio file and hide a message inside. The secret text is embedded via steghide. Once the message has been incorporated, save the audio file and provide it to the challengers. The players must employ steganography methods to decipher the audio file's concealed message. To assist them, you could offer some suggestions or cues. Make sure the message you include in the audio file is not only understandable but also not too evident. To embed the message, use a programme like DeepSound or SilentEye. With the aid of these tools, you are able to encrypt the message and incorporate it into the audio file without affecting the audio's quality.

c) Conclusion: In short, the goal of this stenography challenge is to use stenography to conceal a message within an audio file. The concealed message must be found using stenography skills, and participants must submit it as their response to the challenge. This challenge may be adjusted to be as simple or complex as you like with a little imagination and talent, making it a perfect method to gauge your participants' stenography prowess

Hosting the CTF

Since you will need to safeguard the website from brute force and DDoS attacks, hosting a CTF might be challenging. However, if you have experience managing websites and setting up servers, it is possible. CTFd is a well-known open source CTF hosting platform that is simple to use and includes effective tools for managing the event. There is a plethora of information available in documents, articles, and videos on how to set up CTFd, but we won't discuss it in the paper because it isn't on our learning agenda

III. CONCLUSION

We discussed the various kinds of CTF challenges that may be built and went into great detail about the resources and knowledge that are needed. Hopefully, this study paper assisted you in creating CTF problems, provided you with useful insight into how to think like a challenge.

IV. ACKNOWLEDGEMENT

I take this opportunity to acknowledge everyone who have helped us in every stage of this project. Firstly, I am indebtedly grateful to our Guide and HoD of Cyber Security Department Dr. Nilakshi Jain and Co-Guide Ms. Pranali Pawar for their support. Without their support this project would not have been completed.

Secondly, I would like to thank my group member Deep Parasiya and Chirag Prajapati for their contribution to the project. Also, I would like to thank all the faculty members of our school/college for their kindness and support.

Lastly, I should really thank my friends and family who were always there to support me whenever needed.

REFERENCES

[1] R. Raman, S. Sunny, V. Pavithran and K. Achuthan, "Framework for evaluating Capture the Flag (CTF) security competitions," International Conference for Convergence for Technology-2014, Pune, India, 2014, pp. 1-5, doi: 10.1109/I2CT.2014.7092098.

[2] S. Sharma, R. Kumar, A. Jain, B. Agarwal and S. K. Suman, "Intensifying Practical Based Learning of Penetration Testing using CTF," 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2021, pp. 1378-1381, doi: 10.1109/ICAC3N53548.2021.9725762.

[3] M. N. Uddin, S. Mishra, A. Das, T. Rastogi, A. Verma and P. Kothari, "An Implementation of Capture the Flag (CTF) Tool," 2021 International Conference on Technological Advancements and Innovations (ICTAI), Tashkent, Uzbekistan, 2021, pp. 388-393, doi: 10.1109/ICTAI53825.2021.9673160.

[4] Q. Yan, W. Lai and Z. Wang, "Online Experiments Based on the CTF Model for Information Security MOOC Courses," 2021 16th International Conference on Computer Science & Education (ICCSE), Lancaster, United Kingdom, 2021, pp. 783-788, doi: 10.1109/ICCSE51940.2021.9569691.

[5] K. Zhang, S. Wuthier, K. Yoon and S. -Y. Chang, "Designing and Using Capture The Flag for Coordination and Interaction in Engineering Education," 2022 IEEE Global Engineering Education Conference (EDUCON), Tunis, Tunisia, 2022, pp. 1555-1560, doi: 10.1109/EDUCON52537.2022.9766724.

[6] Á. Balogh, M. Érsok, L. Erdődi, A. Szarvák, E. Kail and A. Bánáti, "Honeypot optimization based on CTF game," 2022 IEEE 20th Jubilee World Symposium on Applied Machine Intelligence and Informatics (SAMI), Poprad, Slovakia, 2022, pp. 000153-000158, doi: 10.1109/SAMI54271.2022.9780835.

[7] S. Roschke, C. Willems and C. Meinel, "A security laboratory for CTF scenarios and teaching IDS," 2010 2nd International Conference on Education Technology and Computer, Shanghai, China, 2010, pp. V1- 433-V1-437, doi: 10.1109/ICETC.2010.5529213.

[8] Z. Liu, H. Qiu, J. Zhu and Z. Zeng, "AAG: A Model for Attack Behavior Judgment in CTF-style Cyber Security Training," 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2019, pp. 1-5, doi: 10.1109/ICSESS47205.2019.9040727.

[9] K. H. Tan and E. L. Ouh, "Lessons Learnt Conducting Capture the Flag CyberSecurity Competition during COVID-19," 2021 IEEE Frontiers in Education Conference (FIE), Lincoln, NE, USA, 2021, pp. 1-9, doi: 10.1109/FIE49875.2021.9637404.

[10] Z. Romano, J. Windsor, M. VanDerPol and J. Coffman, "Election Security in the Cloud: A CTF Activity to Teach Cloud and Web Security," 2021 IEEE Frontiers in Education Conference (FIE), Lincoln, NE, USA, 2021, pp. 1-5, doi: 10.1109/FIE49875.2021.9637368.

[11] H. Huang, J. Ding, W. Zhang and C. J. Tomlin, "Automation-Assisted Capture-the-Flag: A Differential Game Approach," in IEEE Transactions on Control Systems Technology, vol. 23, no. 3, pp. 1014- 1028, May 2015, doi: 10.1109/TCST.2014.2360502.

[12] D. Szedlak and A. M'manga, "Eliciting Requirements for a Studentfocussed Capture The Flag," 2020 7th International Conference on Behavioural and Social Computing (BESC), Bournemouth, United Kingdom, 2020, pp. 1-4, doi: 10.1109/BESC51023.2020.9348329.

[13] R. Singh, "Software Security (Capture the Flag)," 2021 Fourth International Conference on Computational Intelligence and Communication Technologies (CCICT), Sonepat, India, 2021, pp. 165- 168, doi: 10.1109/CCICT53244.2021.00041.

[14] K. Chain, C. -C. Kuo, I. -H. Liu, J. -S. Li and C. -S. Yang, "Design and implement of capture the flag based on cloud offense and defense platform," 2018 IEEE International Conference on Applied System Invention (ICASI), Chiba, Japan, 2018, pp. 686-689, doi: 10.1109/ICASI.2018.8394350.

[15] T. Powers, M. Novitzky and C. Korpela, "Improving Reward Functions in Robots Playing Capture the Flag Using Q-Learning," 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), NV, USA, 2021, pp. 0426-0431, doi: 10.1109/CCWC51732.2021.9375906