Cybersecurity Internship Report Intern Name: vishakha ghadage And Mansi Jagtap. Supriya thamke

Program: Digisuraksha Parhari

Foundation Internship Issued By: Digisuraksha Parhari Foundation

Supported By: Infinisec Technologies Pvt. Ltd.

🧩 Project Report: OverTheWire – Krypton Wargame Analysis

🔰 Objective:

The Krypton wargame introduces beginners to the world of classical cryptography. The goal is to solve

each level by decoding encrypted text and retrieving passwords to access the next level.

🧩 Tools Used:
 Linux Terminal (SSH for remote access)
 Base64 decoding tool
 Caesar cipher decoder
 Online tools: CyberChef, dCode
 Text editors: nano, vim, cat, grep

Task: Solve the Krypton Level 0

Lab Overview:

Lab Name: Krypton

Level: 0 (Base64 Encoding)

Objective: Decode the Base64 string to obtain the password for login.

✅ Level-by-Level Breakdown:

Approach and Steps:

1. Accessing the Lab:

Opened the OverTheWire website and accessed the Krypton game. The challenge begins by decoding a Base64 string to retrieve the password.

2. Base64 String Provided:

The encoded string provided in the challenge was:

S1JZUFRPTklTR1JFQVQ=

3. Decoding Process:

I used CyberChef to decode the Base64 string:

CyberChef Steps:

1. Opened CyberChef.

2. Pasted the encoded string in the Input

section.

3. Applied the From Base64 operation.

The decoded output was:

KRYPTONISGREAT

4. Verification of Output:

The decoded password is KRYPTONISGREAT.

5. Login to Krypton Server:

The next step was logging in to the Krypton server using the SSH command:

ssh krypton1@krypton.labs.overthewire.org -p 2231

When prompted for the password, I entered KRYPTONISGREAT.

Commands Used:

echo "S1JZUFRPTklTR1JFQVQ=" | base64 --decode
ssh krypton1@krypton.labs.overthewire.org -p 2231
KRYPTONISGREAT

This is the decoded password from the Base64 string S1JZUFRPTklTR1JFQVQ=.

So, after running the command:

echo "S1JZUFRPTklTR1JFQVQ=" | base64 --decode

The result you get in the terminal will be:

KRYPTONISGREAT

🔐 Krypton Level 1 → Level 2

Challenge:

 A message is encrypted with a Caesar

cipher (ROT13 logic).

Logic:

 ROT13 shifts each letter by 13 places.

Command used:

cat /krypton/krypton1/keyfile.dat | tr 'A-Za-z' 'N-ZA-Mn-za-m'

Explanation:

 tr command is used to perform letter substitution for ROT13.

🔐 Krypton Level 2 → Level 3

Challenge:

 Ciphered text in a file

 Hint that the text was shifted and characters can only be uppercase

Steps:

1. View the file:

2. cat /krypton/krypton2/keyfile.dat

3. Use Caesar cipher brute force:

o Try all 26 shifts (or use dCode's Caesar solver)

4. Identify plaintext line and extract password.

🔐 Krypton Level 3 → Level 4

Challenge:

 Encrypted password using monoalphabetic substitution cipher

Steps:

1. Analyze the cipher file:

2. cat /krypton/krypton3/keyfile.dat

3. Frequency analysis:

o Replace common letters based on English frequency (E, T, A, O...)

o Use online tools like dCode or do manual mapping.

Tools:
 CyberChef (Frequency Analysis)

 Python (for mapping logic, optional)

🔐 Krypton Level 4 → Level 5

Challenge:

 Similar monoalphabetic cipher, but encoded using a key-based substitution

Logic:

 Custom key provided in the script: THEQUICKBROWNFXJMPSVLAZYDG

 Remaining letters filled alphabetically

Python or bash used to decode:

 Create two strings (plain and cipher) and map using tr

🧠 What We Learned:

 Basics of classical ciphers: Caesar, Substitution, Base64, Vigenèr
 How to use Linux tools (tr, cat, base64, ssh)

 Importance of pattern recognition and frequency analysis

 Practiced reading, decoding, and logical thinking

📝 Conclusion:

The Krypton wargame builds a strong foundation in cryptography and Linux command-line usage. It’s

beginner-friendly yet intellectually engaging — perfect for those stepping into

cybersecurity or ethical

hacking.

Certainly! Here's a comprehensive report analyzing the OverTheWire Natas wargame levels 0 through 10.

This report includes step-by-step solutions, tools utilized, and the underlying logic for each level.

🔐 OverTheWire Natas Levels 0–34:

Comprehensive Analysis

Objective: The Natas wargame is designed to teach the basics of server-side web security by challenging

players to find passwords hidden in web pages, source code, cookies, and more

🔷 Level 0: Introduction to Viewing Page Source

 *URL: http://
natas0.natas.labs.overthewire.or

 *Username: natas

 *Password: natas

Approach:

Upon accessing the page, it displays: "You can find the password for the next level on this page"

However, the password isn't visible in the rendered conten.

 *Solution: View the page source (Right-click → "View Page Source") to find the password within

an HTML commen.

*Tools Used: Web browser's "View Page Source" featur.

*Logic: Understanding that developers may leave sensitive information in HTML comments, which are

not rendered on the page but are visible in the source cod.

🔷 Level 1: Exploring Developer Tools

 *URL: http://
natas1.natas.labs.overthewire.og

 *Username: nata1

 *Password: (Obtained from Level0)

Approach:

 The page displays: "You can find the password for the next level on this pag." - No visible

password in the rendered contet.

 *Solution: Use browser developer tools (F12) to inspect the page elements and find the password

within an HTML commet.

*Tools Used: Browser Developer Tools (Elements ta).

*Logic: Recognizing that inspecting the DOM can reveal hidden elements or comments containing

sensitive informatin.

🔷 Level 2: Directory Enumeration

 URL http://natas2.natas.labs.overthewire.rg

 Username nats2

 Password (Obtained from Leve 1)

Approach:

 The main page doesn't provide any useful informaton.

 Solution View the page source to find a reference to an "images" directry.

 Navigate to http://natas2.natas.labs.overthewire.org/files/ to find a directory listng.

 Locate a file containing the passwrd.

Tools Used Web browser for manual directory traveral.

Logic Understanding that web servers may have accessible directories that are not linked on the main page

but can be discovered through hints in the source cde.

🔷 Level 3: Utilizing robots.txt

 UR: http://natas3.natas.labs.overthewireorg

 Usernam: naas3

 Passwor: (Obtained from Levl 2)

*Approach:

 The main page provides no useful informaion.

 Solutio: Access http://natas3.natas.labs.overthewire.org/robots.txt to find disallowed

directoies

 Navigate to the disallowed directory to find a file containing the passord.

Tools Use: Web browser to access robots.txt and disallowed directoies.

Logi: Recognizing that robots.txt files can reveal hidden directories that are intended to be excluded

from search engine indexing but may contain sensitive informaion.

🔷 Level 4: Referer Header Manipulation

 UL: http://
natas4.natas.labs.overthewir.org

 Usernae: ntas4

 Passwod: (Obtained from Leel 3)

*Approach:

 Accessing the page results in a message denying acess.

 Solutin: Use browser developer tools to modify the "Referer" header in the HTTP request to

`http://natas5.natas.labs.overthewireorg.

 Refresh the page with the modified header to gain access and retrieve the

pasword.

Tools Usd: Browser Developer Tools (Network tab) or tools like cURL to modify HTTP heders.

Logc: Understanding that some web applications use the "Referer" header for access control, which can be

manipulated to bypass restricions.

🔷 Level 5: Cookie Manipulation

 RL: http://
natas5.natas.labs.overthewie.org

 Usernme: atas5

 Passwrd: (Obtained from Lvel 4)

Approach

 The page indicates that access is enied.

 Soluton: Inspect the cookies set by the server using browser developertoos.

 Find a cookie named loggedin set o ``.

 Change its value to 1 and refresh the page to gain access and retrieve the pasword.

Tools Ued: Browser Developer Tools (Application tab) to view and modify cokies.

Loic: Recognizing that authentication states can be stored in cookies, which, if not securely implemented,

can be manipulated to gain unauthorized ccess.

🔐 OverTheWire Leviathan:

Comprehensive Walkthrough

🔷 Level 0 → 1

 *Objective: Locate the password for `leviathan1.

 *Tools Used: ls, cd, cat, gre

 Solution Logic: 1 Access the .backup directory using cd .backup. 2 Identify the

bookmarks.htmlfil. 3 Usegrep` to search for the keyword "password" within the fil:

 grep "password" bookmarks.html

4 Extract the password from the line containing i.

 *Password: rioGegei8

🔷 Level 1 → 2

 *Objective: Discover the password for leviathan.

 *Tools Used: file, ltrae

 Solution Logic: . Identify the check binary using l. . Confirm it's an executable with file

chec. . Use ltrace to trace library calls and identify the password comparisn:

 ltrace ./check

. Note the password it compares against and use it to execute the binay.

 *Password: ougahZi8a

🔷 Level 2 → 3

 Objective Gain access to leviatha3.

 Tools Used ltrace, touch, mkdir, bsh

 Solution Logic:

1. Identify the printfile binry.
2. Use ltrace to observe its behavor:

3. ltrace ./printfile

4. Create a file with a name that includes a command injection, such as test; bah.

5. Execute the binary with the crafted filename to spawn a shll.

 Password Ahdiemo1j

🔷 Level 3 → 4

 Objectiv: Retrieve the password for leviathn4.

 Tools Use: ltace

 Solution Logic:

1. Identify the level3 biary.

2. Use ltrace to monitor its behavior and identify the password it expcts:

3. ltrace ./level3

4. Input the correct password to gain acess.

 Passwor: vuH0cox6m

🔷 Level 4 → 5

 Objectie: Find the password for leviatan5.

 Tools Usd: ls, cd, ./bin, binary-to-ASCII convrsion

 Solution Logic: 1. Navigate to the .trash diretory. 2. Execute the bin file to obtain binary

otput. 3. Convert the binary output to ASCII to reveal the pasword.

 Passwod: Tith4okei

🔷 Level 5 → 6

 Objectve: Access leviahan6.

 Tools Ued: ln, symboli links

 *Solution Logic:

1. Create a symbolic link in /tmp pointing to the passwor file:

2. ln -s /etc/leviathan_pass/leviathan6 /tmp/file.log

3. ``

4. Execute the leviathan5 binary, which reads from /tmp/file.log, to display the

pasword.

 Passwrd: Ugaoee4li

🔷 Level 6 → 7

 Objecive: Obtain the password for levithan7.

 Tools sed: Python scripting, brute-force pproach

 *Solution Logic:

1. Develop a Python script to brute-force the 4-digit PIN required by the leviathan6binary
2. Iterate through possible combinations until the correct PIN is found and access is

ranted.

 Passord: ahyMaeBo9

🔷 Level 7 → 8

 Objetive: Complete the finl level.

 ToolsUsed: strings, grep`

 Solution Logic

1. Use strings to extract readable strings from the `leviathan7 binary:

2. strings leviathan7

3. ``

4. Identify any hardcoded passwords or hints within th outpu.

5. Use the discovered information to gai access.

 Pasword: loVzZ6mT