DAILY ONLINE ACTIVITIES SUMMARY

Date:	**20-05-2020**	**Name:**	**Manikya K**
Sem & Sec	**8th,A**	**USN:**	**4AL16CS050**

Online Test Summary

Subject	**IOT(IA1)**		
Max. Marks	**30**	**Score**	**23**

Certification Course Summary

Course	**Introduction to ethical hacking**		
Certificate Provider	**Great learner academy**	**Duration**	**6 Hrs**

Coding Challenges

Problem Statement: prob1- Write a C Program to Reverse a Linked List in groups of given size.

Test Case 1:
If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8
The value of size k is 2
Then the linked list looks like: 2 → 1 → 4 → 3 → 6 → 5 → 8 → 7

Test Case 2:
If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8
The value of size k is 3
Then the linked list looks like: 3 → 2 → 1 → 6 → 5 → 4 → 8 → 7

Status: Solved

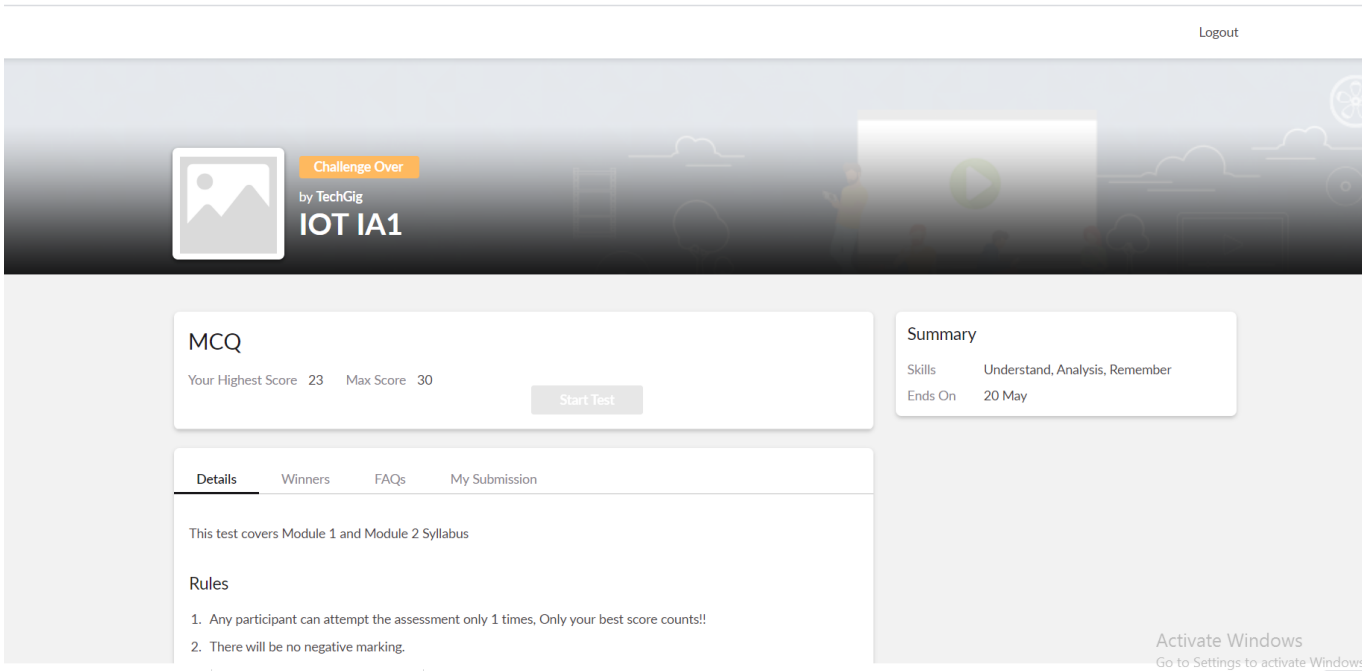
Uploaded the report in Github	**Yes**
If yes Repository name	**manikya-20**
Uploaded the report in slack	**Yes**

Online Test Details: (Attach the snapshot and briefly write the report for the same)

Certification Course Details: (Attach the snapshot and briefly write the report for the same)

Coding Challenges Details: (Attach the snapshot and briefly write the report for the same)

1) Online Test Details:



2) Certification Course Details:

In the dawn of international conflicts, terrorist organizations funding cybercriminals to breach security systems, either to compromise national security features or to extort huge amounts by injecting malware and denying access. Resulting in the steady rise of cybercrime. Organizations face the challenge of updating hack-preventing tactics, installing several technologies to protect the system before falling victim to the hacker.

New worms, malware, viruses, and ransomware are multiplying every day and is creating a need for ethical hacking services to safeguard the networks of businesses, government agencies or defense.

greatlearning
Learning for Life

Introduction to Ethical Hacking

CONTENT ASSESSMENTS

Learning Videos

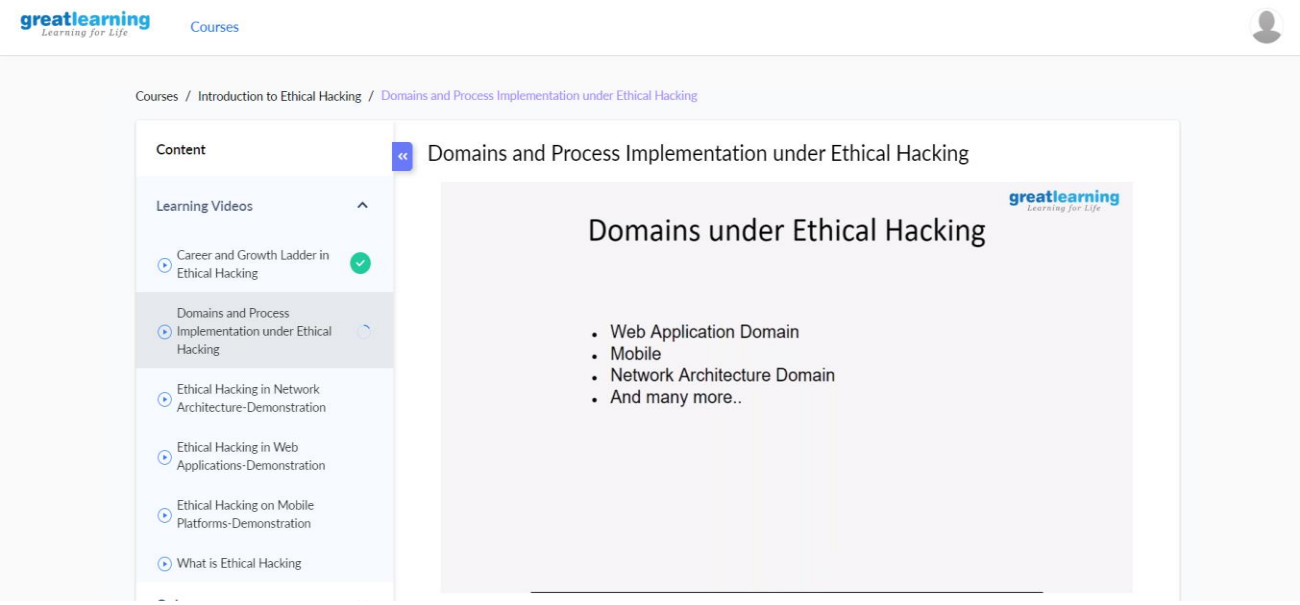
- Career and Growth Ladder in Ethical Hacking — 18m
- Domains and Process Implementation under Ethical Hacking — 54m
- Ethical Hacking in Network Architecture-Demonstration — 48m
- Ethical Hacking in Web Applications-Demonstration — 50m
- Ethical Hacking on Mobile Platforms-Demonstration — 34m
- What is Ethical Hacking — 50m

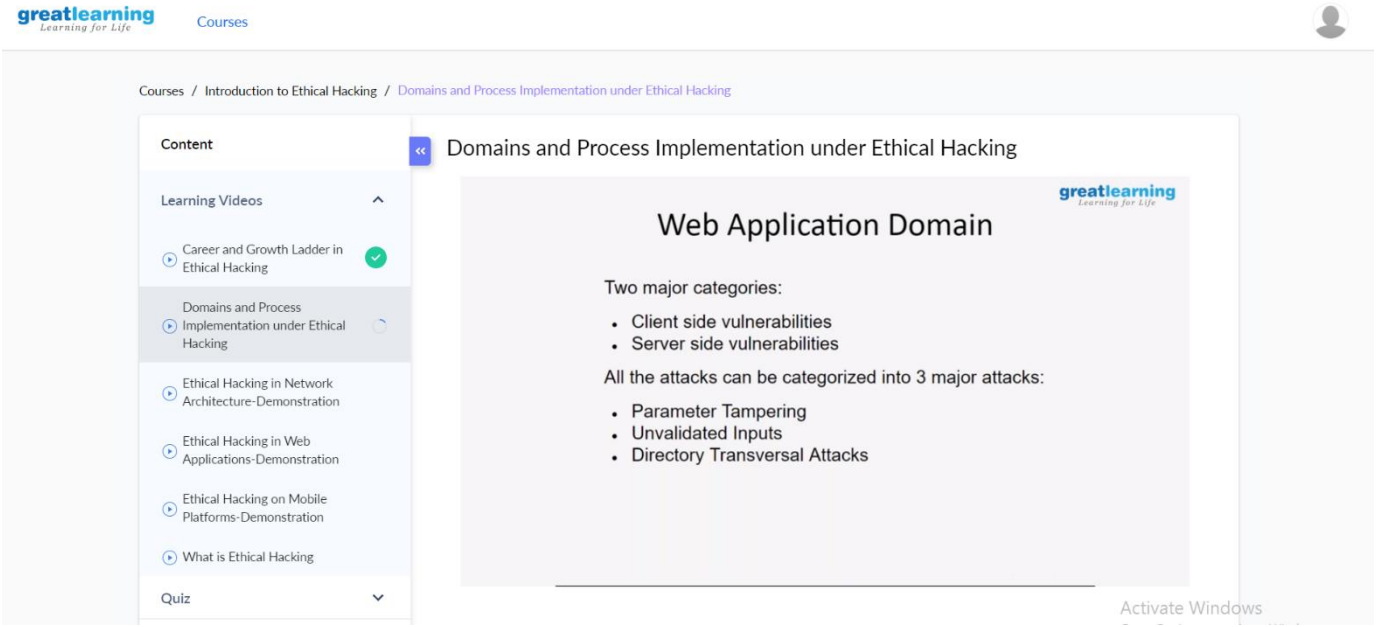
Quiz

- Ethical Hacking - Quiz

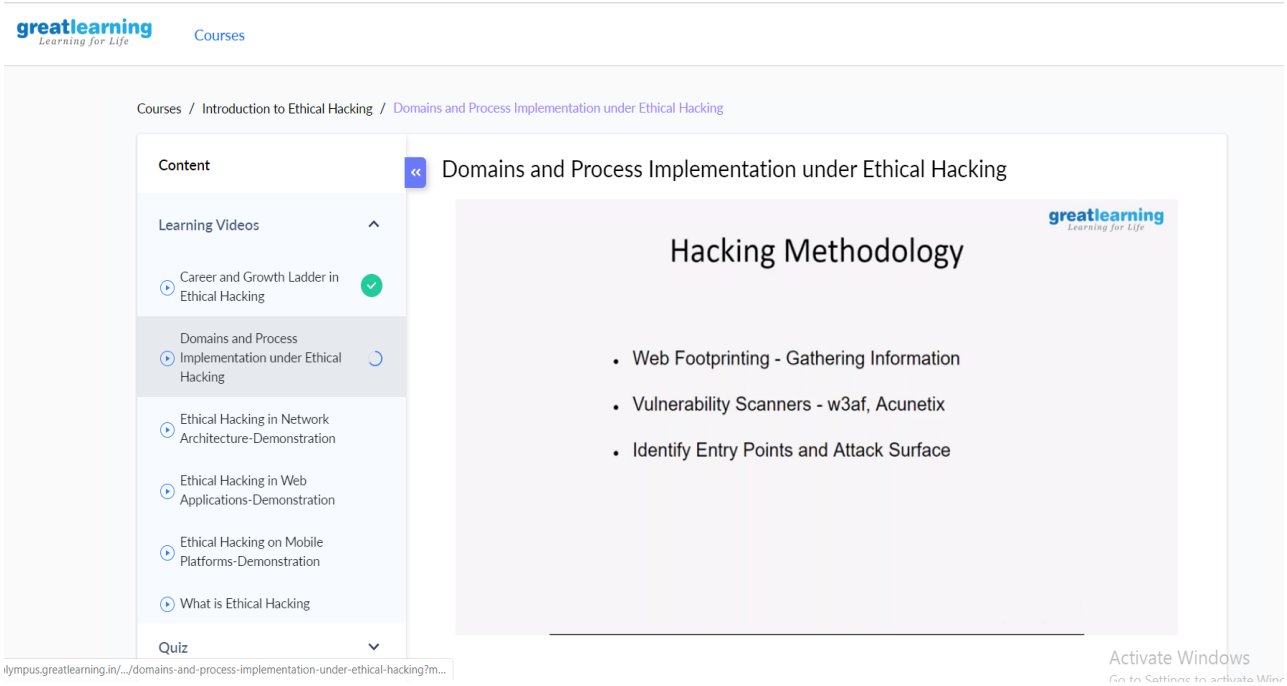
Claim Your Course Certificate

- Claim your course certificate





There are many strategies for training your employees. The overall practice is called White Hat Social Engineering. A certified ethical hacker uses the very same techniques that the bad guys will use to train employees on what to look for, scaring them a little bit. Now we don't want them to be so paranoid that it actually effects their productivity. But there's a nice balance you need to establish with your users between basic awareness and productivity.



Although there is no specific step-by-step methodology used by all hackers, a typical hacking process comprises of the following steps:

1. **Footprinting** – the process of using passive methods of gaining information about the target system prior to performing the attack. The interaction with the target system is kept at minimum in order to avoid detection and alert the target about the attack. The footprinting can reveal vulnerabilities of the target system and improve the ease with which they can be exploited. Various methods are employed for footprinting, for example whois queries, Google searches, job boards search, network enumeration, operating system identification, etc.
2. **Scanning** – the process of taking information obtained from the footprinting phase in order to target the attack more precisely. Some of the metods used in this phase are port scans, ping sweeps, operating systems detection, observation of facilities used by the target, and so on.
3. **Enumeration** – the process of extracting more detailed information about the information obtained during the scanning phase to determine its usefulness. Some of the methods used in this step are user accounts enumeration, SNMP enumeration, UNIX/Linux enumeration, LDAP enumeration, NTP enumeration, SMTP enumeration, DNS enumeration, etc.
4. **System hacking** – the process of planning and executing the attack based on the information obtained in the previous phases. In this phase the attacker performs the actual hacking process using hacking tools.
5. **Escalation of privilege** – the process of obtaining privileges that are granted to higher privileged accounts than the attacker broke into originally. The goal of this step is to move from a low-level account (such as a guest account) all the way up to administrator.

3)Coding Challenges Details:

Write a C Program to Reverse a Linked List in groups of given size.

Test Case 1:
If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8
The value of size k is 2
Then the linked list looks like: 2 → 1 → 4 → 3 → 6 → 5 → 8 → 7

Test Case 2:
If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8
The value of size k is 3
Then the linked list looks like: 3 → 2 → 1 → 6 → 5 → 4 → 8 → 7

```
#include<stdio.h>
#include<stdlib.h>
struct Node
{
int data;
struct Node* next;
};
pointer to the new head node. /
struct Node reverse (struct Node head, int k)
{
struct Node current = head;
struct Node next = NULL;
struct Node prev = NULL;
int count = 0;
while (current != NULL && count < k)
{
next = current->next;
current->next = prev;
prev = current;
current = next;
count++;
}

if (next != NULL)
head->next = reverse(next, k);

return prev;
}
void push(struct Node** head_ref, int new_data)
{
struct Node* new_node =
```

```
(struct Node*) malloc(sizeof(struct Node));
new_node->data = new_data;

new_node->next = (*head_ref);

(*head_ref) = new_node;
}
void printList(struct Node node)
{
while (node != NULL)
{
printf("%d ", node->data);
node = node->next;
}
}
int main(void)
{
struct Node head = NULL;
push(&head, 8);
push(&head, 7);
push(&head, 6);
push(&head, 5);
push(&head, 4);
push(&head, 3);
push(&head, 2);
push(&head, 1);
printf("\nGiven linked list \n");
printList(head);
head = reverse(head, 2);

printf("\nReversed Linked list \n");
printList(head);

return(0);
```