TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**Project 3**

**TÌM HIỂU VỀ WINDOWS SERVER**

**VÀ XÂY DỰNG CHƯƠNG TRÌNH**

**LẤY THÔNG TIN TỪ ACTIVE DIRECTORY**

Sinh viên thực hiện: **Lê Ngọc Tuấn**

Lớp CN-CNTT 02 - K58

Giáo viên hướng dẫn: Th.S. **Nguyễn Đức Tiến**

Hà Nội, tháng 05 năm 2016

# PHIẾU GIAO NHIỆM VỤ MÔN HỌC

**1. Thông tin về sinh viên**

Họ và tên sinh viên: Lê Ngọc Tuấn

Điện thoại liên lạc: 01695559591 Email: ngoctuanle020395@gmail.com

Lớp: CN - CNTT 02 - K58 Hệ đào tạo: Đại học chính quy

**2. Sản phẩm mục tiêu của môn học**

- TÌM HIỂU VỀ WINDOWS SERVER VÀ XÂY DỰNG CHƯƠNG TRÌNH LẤY THÔNG TIN TỪ ACTIVE DIRECTORY

**3. Các tính năng**

**Các tính năng bắt buộc:**

* Lập trình/Tìm hiểu để hoàn thành sản phẩm: Có
* Sử dụng một kho repository để lưu trữ các phiên bản của project: Có
* https://github.com/ngoctuanle/ActiveDirectoryInfomationConsole

**Các tính năng tùy chọn:**

* Đăng nhập với tài khoản đã có ở phía máy chủ.
* Lấy thông tin về User, OU, Groups, Computer từ Active Directory.
* Sử dụng Attributes để phân quyền cho từng OU khác nhau.

Table of Contents

[I. Tổng quan đề tài: 3](#_Toc450739292)

[1.1 Mô tả yêu cầu: 3](#_Toc450739293)

[1.2 Mục tiêu: 3](#_Toc450739294)

[1.3 Phương pháp triển khai và công nghệ sử dụng 3](#_Toc450739295)

[II. Cài đặt hệ thống 3](#_Toc450739296)

[2.1 Cài đặt Active Directory Domain Serivce trên Windows Server 2012 3](#_Toc450739297)

[2.2 Cài đặt DHCP trên Server và join Domain từ máy client 5](#_Toc450739298)

[III. Xây dựng chương trình để lấy thông tin từ ActiveDirectory 8](#_Toc450739299)

[3.1 Phân tích cách thức hoạt động của chương trình 8](#_Toc450739300)

[3.2 Các hàm chức năng cơ bản của chương trình 9](#_Toc450739301)

[3.2.1 Hàm kiểm tra user có hợp lệ để tham gia chương trình 9](#_Toc450739302)

[3.2.2 Hàm lấy thông tin các OU trong Domain mà Computer này đang tham gia 10](#_Toc450739303)

[3.2.3 Hàm lấy thông tin các User trong Domain mà Computer này đang tham gia 10](#_Toc450739304)

[3.2.4 Hàm lấy thông tin các Groups trong Domain mà Computer này đang tham gia 11](#_Toc450739305)

[3.2.5 Hàm lấy thông tin các Computer trong Domain mà Computer này tham gia 12](#_Toc450739306)

[3.2.6 Sử dụng Attributes để giới hạn các quyền đối với các OU khác nhau 12](#_Toc450739307)

# I. Tổng quan đề tài:

## 1.1 Mô tả yêu cầu:

Tìm hiểu tổng quan, cách cài đặt, cấu hình một máy chủ sử dụng Windows Server. Các tổ chức và hoạt động của Active Directory trong một mạng kết nối theo mô hình Client-Server.

## 1.2 Mục tiêu:

Cài đặt thành công và cấu hình chính xác cho một máy chủ cài đặt Windows Server với một tên miền xác định. Kết nối được các máy tính khác tham vào tên miền đó do máy chủ này quản lý.

Từ các máy client, xây dựng một ứng dụng nhỏ để có thể lấy được các thông tin trong ActiveDirectory của máy chủ, từ đó có thể làm một thao tác liệt kê các máy tính, người dùng, phòng ban... đang tham gia trong một mạng, phân quyền giới hạn cho từng phòng ban tương ứng.

## 1.3 Phương pháp triển khai và công nghệ sử dụng

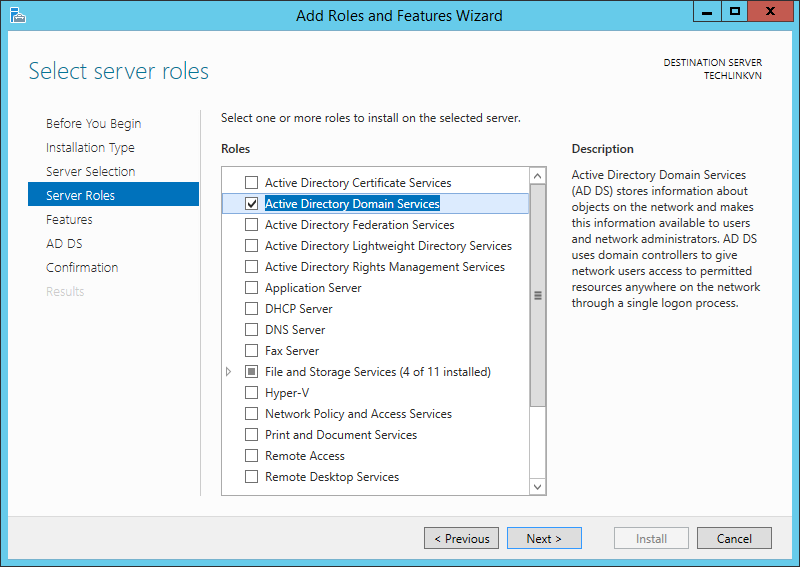
Hệ thống được triển khai với một máy chủ được cài đặt Windows Server 2012 R2 và máy client cài đặt Windows 10.

Ứng dụng nhỏ để lấy thông tin của ActiveDirectory được lập trình trên máy client sử dụng ngôn ngữ C#.

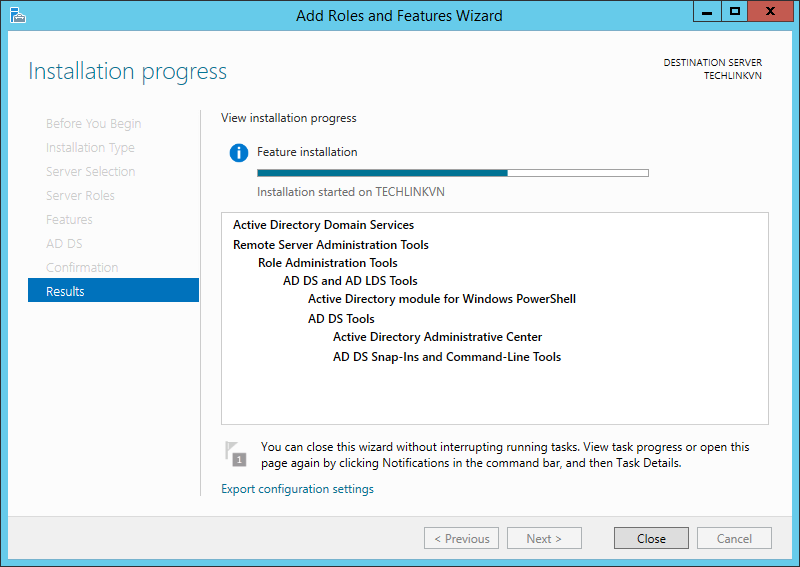
# II. Cài đặt hệ thống

## 2.1 Cài đặt Active Directory Domain Serivce trên Windows Server 2012

Từ cửa sổ Server Manager, ta cài đặt role mới là Active Directory Domain Service như trong hình bên dưới.

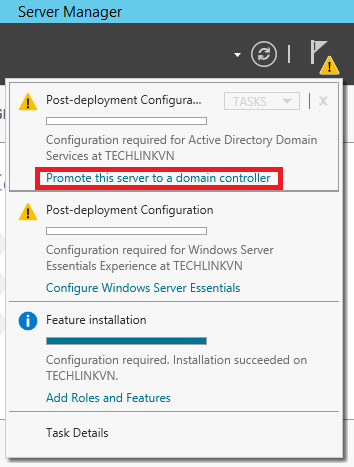


Hình 1 Lựa chọn cài đặt Server Roles

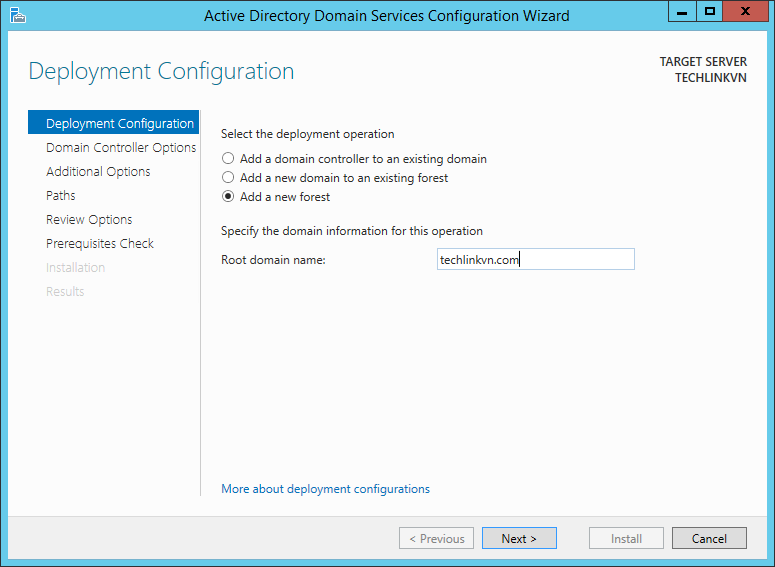
**

Hình 2 Hoàn tất cài đặt Actice Directory Domain Serivce

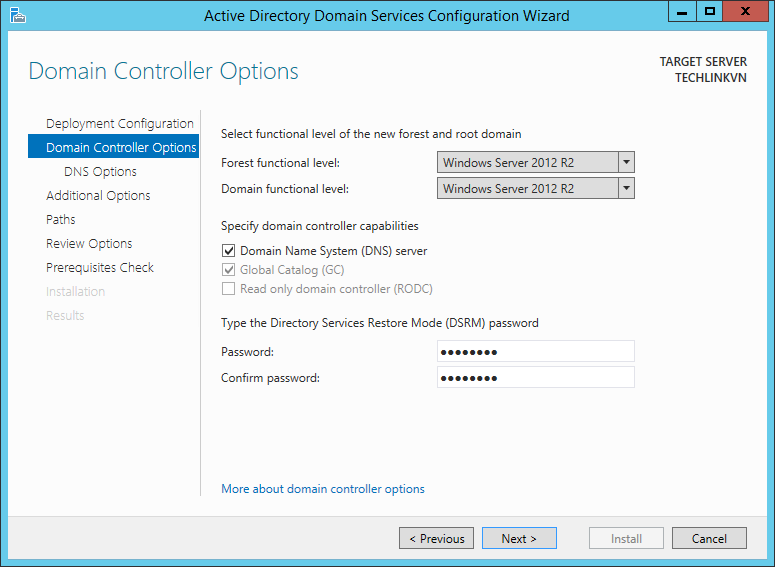
Sau khi thêm role vào Server, ta sẽ Promote Server vào Domain Controller. Thêm tên miền mới, cài đặt Mật khẩu cho Domain Controller.



Hình 3 Promote Server

**

Hình 4 Thêm tên miền mới

**

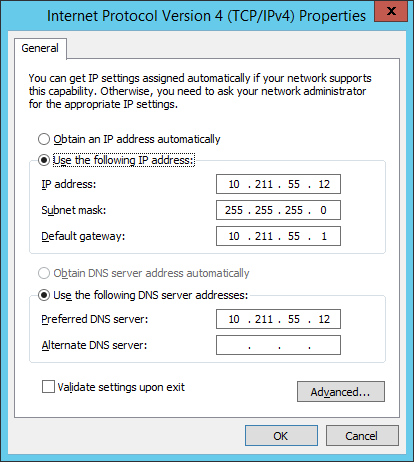
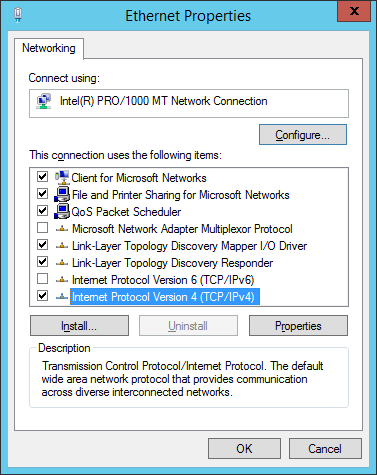
Hình 5 Cài đặt mật khẩu cho Domain Controller

Sau khi Promote Server là ta đã hoàn thành hoạt động cài đặt Active Directory Domain Service, sau bước này hệ thống sẽ khởi động lại, và ta sẽ đăng nhập vào hệ thống dưới một tên miền mới.

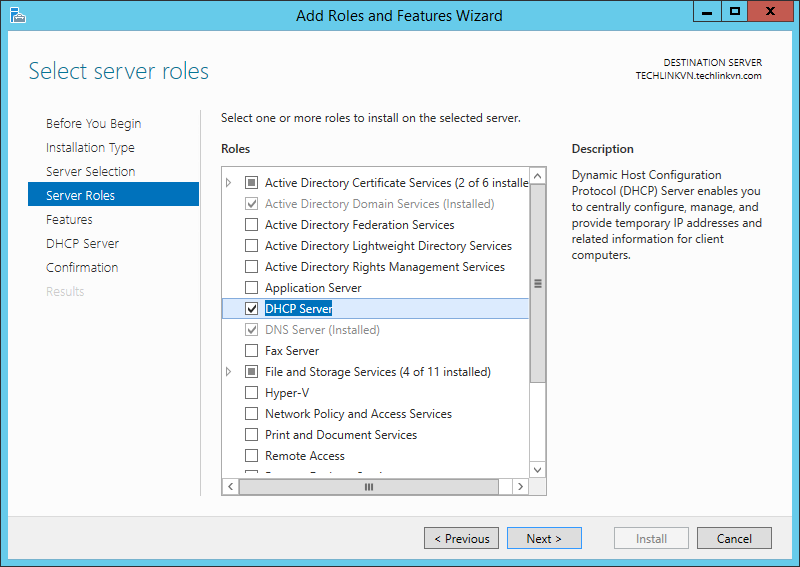
## 2.2 Cài đặt DHCP trên Server và join Domain từ máy client

Tiếp theo ta cần cài đặt DHCP cho máy chủ, để tạo ra một database trung tâm quản lý tất cả các máy tính trong hệ thống mạng.

Để bắt đầu cài đặt DHCP, ta cần cài đặt địa chỉ IP tĩnh cho máy chủ để các máy tính trong mạng có thể dễ dàng kết nối với máy chủ. Tiếp theo đó, ta sẽ thêm một Role mới vào máy chủ có tên là DHCP Server

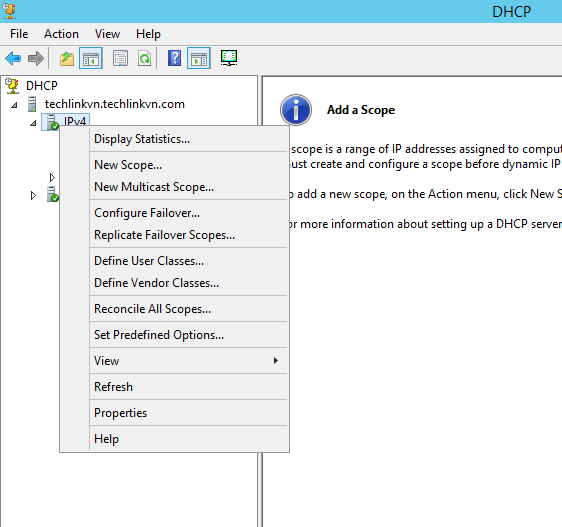


Hình 6 Cài đặt địa chỉ IP tĩnh cho máy chủ

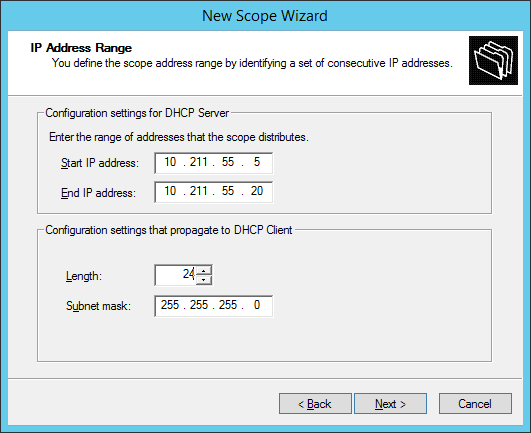


Hình 7 Cài đặt role DHCP Server vào máy chủ

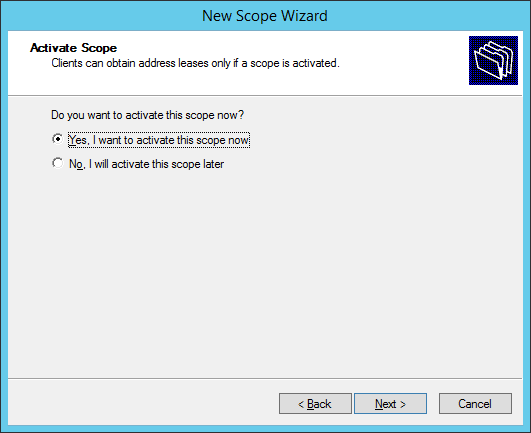
Sau khi cài đặt thành công DHCP vào máy chủ, ta cần thêm một scope vào dải địa chỉ IPv4 mà máy chủ quản lý, để máy chủ có thể lấy các IP trong dải này để cấp phát và quản lý các máy tính kết nối trong mạng.



Hình 8 Thêm dải IP cần quản lý vào máy chủ



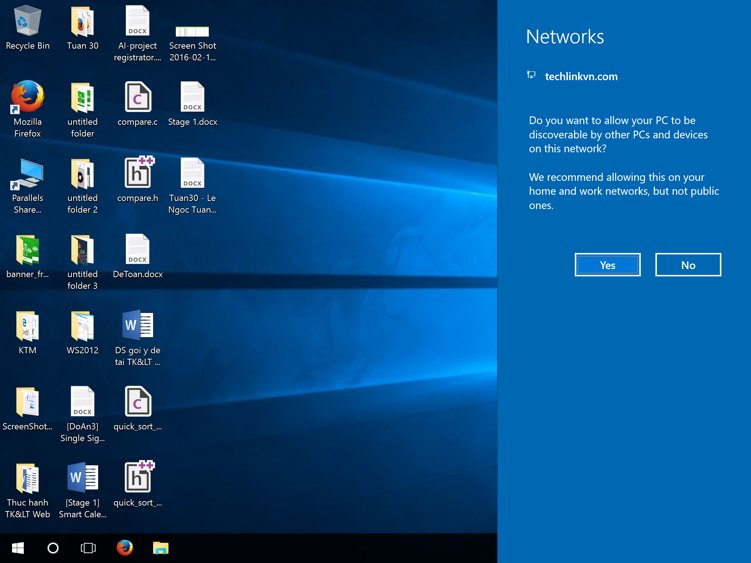
Hình 9 Cài đặt dải địa chỉ IP sẽ cấp phát trong DHCP Server

**

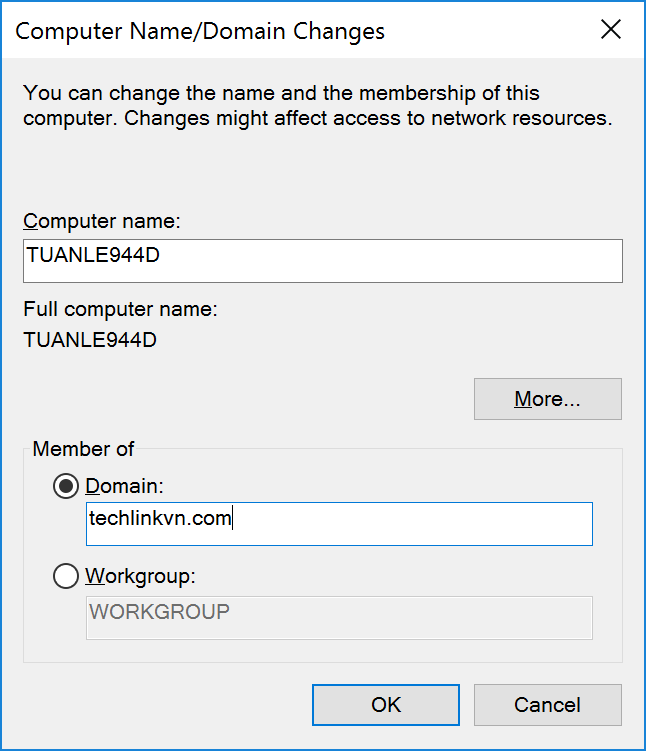
Hình 10 Kích hoạt dải địa chỉ này

Sau khi hoàn thành bước này, ta đã xong việc cài đặt Active Directory Domain Service ở phía Server. Tiếp theo, ta sẽ sử dụng một máy tính kết nối vào mạng này để join vào Domain Controller vừa tạo.

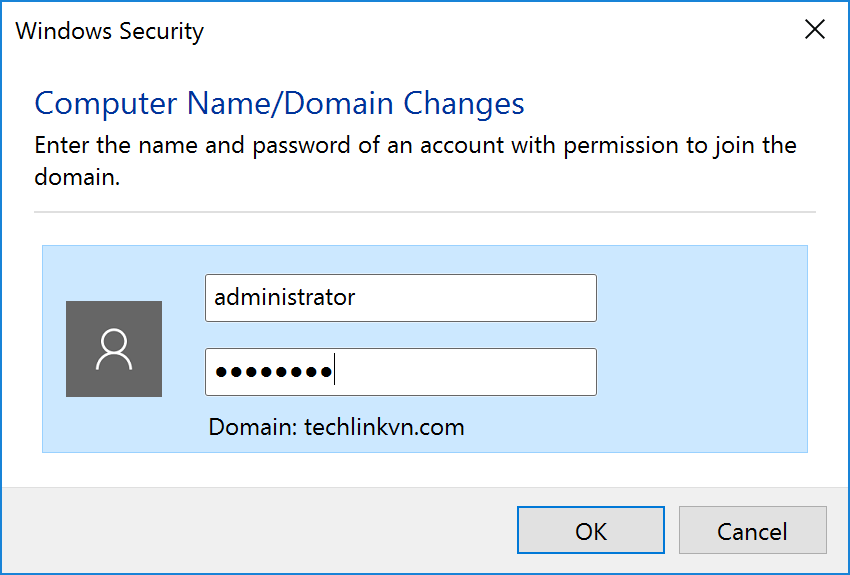
Ngay sau khi kết nối vào mạng và máy tính nằm trong dải địa chỉ mà DHCP quản lý ở bước trên, máy tính sẽ tự động cài đặt vào một mạng mới có tên miền như ta đã cấu hình ở phía máy chủ. Tiếp theo đó ta sẽ join máy tính vào tên miền đã được tạo.



Hình 11 Máy tính được cài đặt vào một mạng có tên miền đã cấu hình

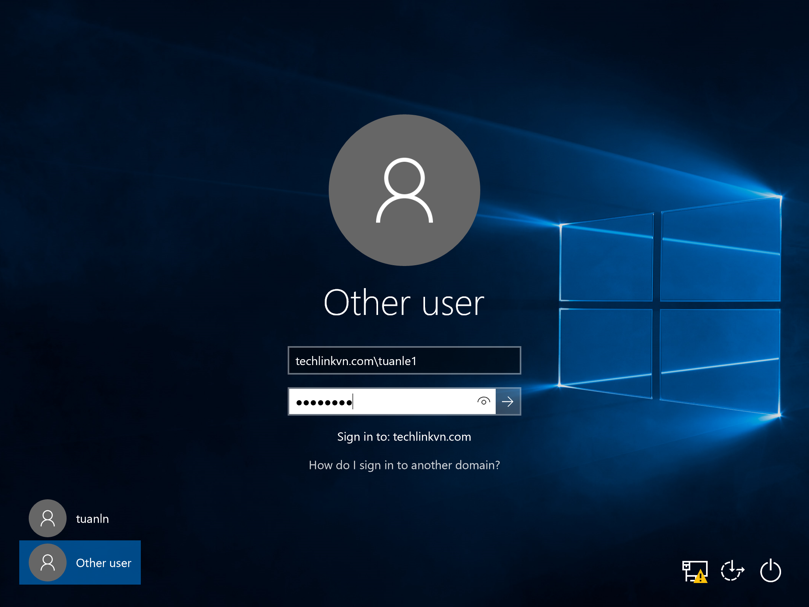
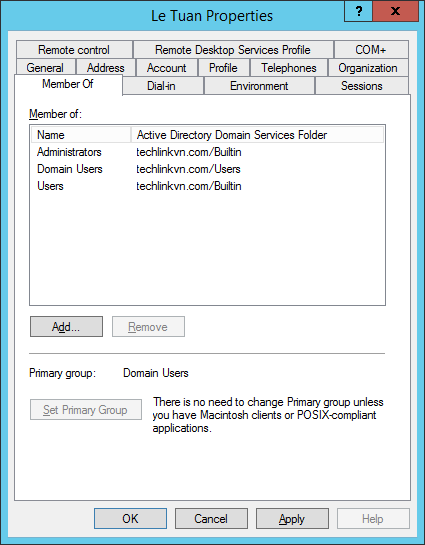


Hình 12 Join Domain

**

Hình 13 Nhập username và mật khẩu của domain controller này để kết nối

Sau bước này, máy tính này sẽ khởi động này và sau đó ta có thể đăng nhập vào máy tính này bằng các tài khoản nằm trong Active Directory đã được tạo vào lưu trữ bên phía máy chủ.



Hình 14 Đăng nhập với tài khoản được tạo phía server

# III. Xây dựng chương trình để lấy thông tin từ ActiveDirectory

## 3.1 Phân tích cách thức hoạt động của chương trình

Sau khi thực hiện việc cài đặt máy chủ và join các máy client vào domain, ta sẽ bắt tay vào thực hiện chương trình nhỏ để có thể lấy được các thông tin từ ActiveDiretory.

Từ những máy tính tham gia trong domain của máy chủ, có thể hoàn toàn liên kết với ActiveDirectory từ máy chủ để lấy được các thông tin về User, OU, Computer nào đang được tham gia trong domain đó.

## 3.2 Các hàm chức năng cơ bản của chương trình

### 3.2.1 Hàm kiểm tra user có hợp lệ để tham gia chương trình

Để chương trình có thể lấy được thông tin từ máy chủ, ta cần phải khai báo hai biến bao gồm cấu trúc của ActiveDirectory mà ta sẽ thực hiện để query các kết quả và một xâu kí tự chứa tên của domain mà máy tính thực hiện chương trình đang tham gia.

// Biến DirectoryEntry để lưu cấu trúc của AD khi thực hiện query

static DirectoryEntry dEntry = null;

//Lưu tên domain mà Computer khi thực hiện chương trình này đang tham gia

private static string stringDomainName = System.Net.NetworkInformation.IPGlobalProperties.GetIPGlobalProperties().DomainName;

Tiếp đó để đảm bảo an toàn, để có thể sử dụng chương trình, người dùng cần đăng nhập với một tài khoản hợp lệ đã được tạo bên phía máy chủ để có thể thực hiện các chức năng của chương trình.

/// <summary>

/// Hàm kiểm tra xem username và password nhập vào có hợp lệ trong domain hay không

/// </summary>

/// <param name="username"></param>

/// <param name="password"></param>

/// <returns>

/// Trả về giá trị bool tương ứng:

/// - True: khi hợp lệ

/// - Falshe: ko hợp lệ

/// </returns>

private static bool validateUser(string username, string password)

{

bool result = true;

try

{

dEntry = new DirectoryEntry("LDAP://" + stringDomainName, username, password);

object nativeObject = dEntry.NativeObject;

} catch (Exception e)

{

dEntry.Dispose();

result = false;

}

return result;

}

### 3.2.2 Hàm lấy thông tin các OU trong Domain mà Computer này đang tham gia

/// <summary>

/// Hàm lấy thông tin các OU trong Domain mà Computer này đang tham gia

/// </summary>

/// <returns>

/// Trả về List chứa các OU

/// </returns>

private static List<string> getOU()

{

List<string> list = new List<string>();

DirectorySearcher dSearcher = new DirectorySearcher(dEntry);

dSearcher.Filter = "(&(objectClass=organizationalUnit))";

foreach(SearchResult result in dSearcher.FindAll())

{

list.Add(TextProcessing.getProperty(result, "ou"));

}

dSearcher.Dispose();

return list;

}

### 3.2.3 Hàm lấy thông tin các User trong Domain mà Computer này đang tham gia

Một User có nhiều thông tin liên quan, nên để có thể biểu diễn một User trong chương trình, ta cần xây dựng một lớp chứa các thuộc tính liên quan đến User đó.

class User

{

/// <summary>

/// Các giá trị lưu trong lớp User tương ứng

/// - SAMAccountName: Tên đăng nhập của User

/// - commonName: tên đầy đủ của User

/// - ou: OU của User

/// </summary>

public string SAMAccountName { get; set; }

public string commonName { get; set; }

public string ou { get; set; }

}

Hàm để lấy ra danh sách các User trong domain

/// <summary>

/// Hàm lấy thông tin các User trong Domain mà Computer này đang tham gia

/// </summary>

/// <returns>

/// Trả về List các User

/// </returns>

private static List<User> getUsers()

{

List<User> list = new List<User>();

DirectorySearcher dSearcher = new DirectorySearcher(dEntry);

dSearcher.Filter = "(&(objectClass=user))";

foreach(SearchResult result in dSearcher.FindAll())

{

list.Add(new User {

SAMAccountName = TextProcessing.getProperty(result, "SAMAccountName"),

commonName = TextProcessing.getProperty(result, "cn"),

ou = TextProcessing.getOU(TextProcessing.getProperty(result, "distinguishedname"))

});

}

return list;

}

### 3.2.4 Hàm lấy thông tin các Groups trong Domain mà Computer này đang tham gia

/// <summary>

/// Hàm lấy thông tin các Groups trong Domain mà Computer này đang tham gia

/// </summary>

/// <returns>

/// Trả về danh sách các Group

/// </returns>

private static List<string> getGroup()

{

List<string> list = new List<string>();

DirectorySearcher dSearcher = new DirectorySearcher(dEntry);

dSearcher.Filter = "(&(objectClass=group))";

foreach (SearchResult sResult in dSearcher.FindAll())

{

list.Add(TextProcessing.getProperty(sResult, "cn"));

}

dSearcher.Dispose();

return list;

}

### 3.2.5 Hàm lấy thông tin các Computer trong Domain mà Computer này tham gia

/// <summary>

/// Hàm lấy thông tin các máy tính trong Domain mà Computer này đang tham gia

/// </summary>

/// <returns>

/// Trả về danh sách các máy tính

/// </returns>

private static List<string> getComputer()

{

List<string> list = new List<string>();

DirectorySearcher dSearcher = new DirectorySearcher(dEntry);

dSearcher.Filter = "(&(objectClass=computer))";

foreach (SearchResult sResult in dSearcher.FindAll())

{

list.Add(TextProcessing.getProperty(sResult, "cn"));

}

dSearcher.Dispose();

return list;

}

### 3.2.6 Sử dụng Attributes để giới hạn các quyền đối với các OU khác nhau

Trong thực tế sử dụng, các phòng ban khác nhau cần được giới hạn các quyền sử dụng để tránh không ảnh hưởng đến các phòng ban khác trong mạng. Chương trình này cũng sẽ nói sơ qua về việc ứng dụng Attributes trong C# để giới hạn.

Trước tiên, ta cần tạo một lớp kết thừa từ lớp System.Attributes với các thuộc tính ta sẽ sử dụng để lấy căn cứ giới hạn các quyền từ phòng ban, cụ thể ở đây là thuộc tính OUname.

[System.AttributeUsage(System.AttributeTargets.Method, AllowMultiple = true)]

class CustomAttributes : System.Attribute

{

private string OUname;

public CustomAttributes(string OUname)

{

this.OUname = OUname;

}

public string getOUname()

{

return OUname;

}

}

Sau đó, để giới hạn các quyền truy cập của các hàm cần thiết, ta chỉ cần đặt một Attributes như sau ở trước hàm đó.

[CustomAttributes("BM KTMT")]

private static List<string> getComputer()

{

...

}