

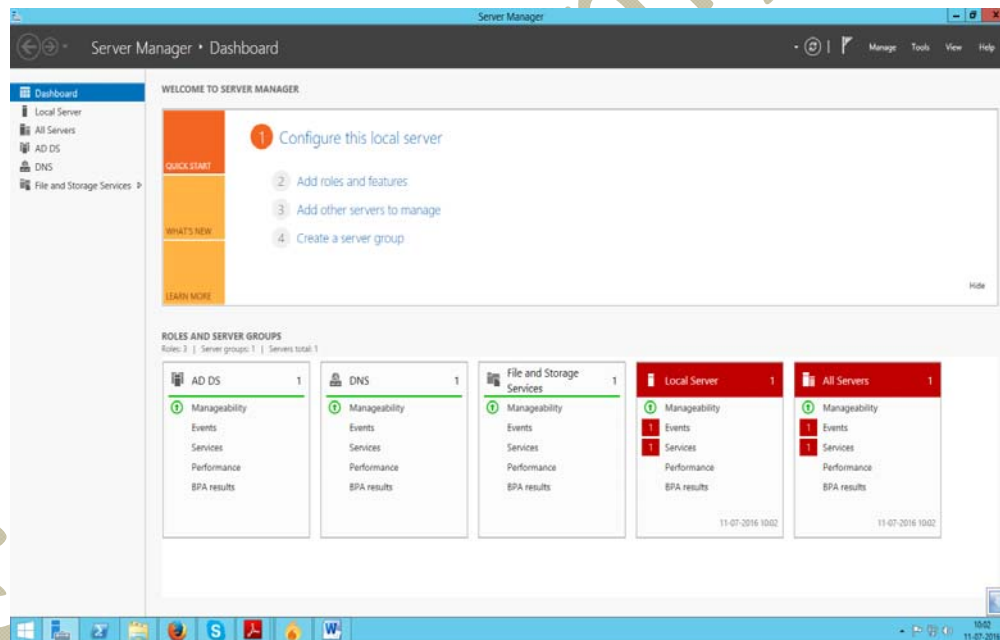


Cloud Computing

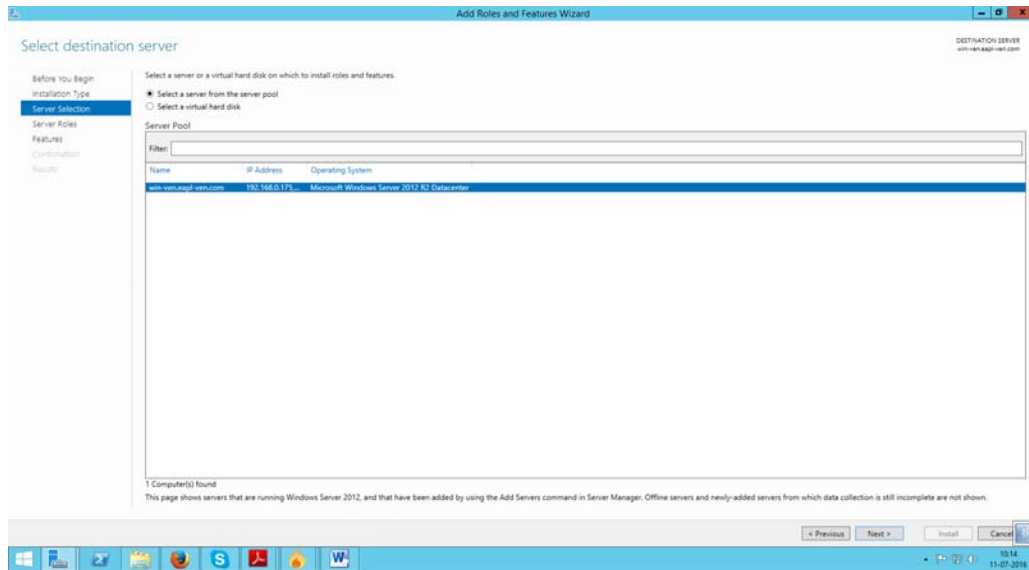
Experiment Number 1.

Procedure to Install and Configure Hyper – V for Creation of Virtual Machines.

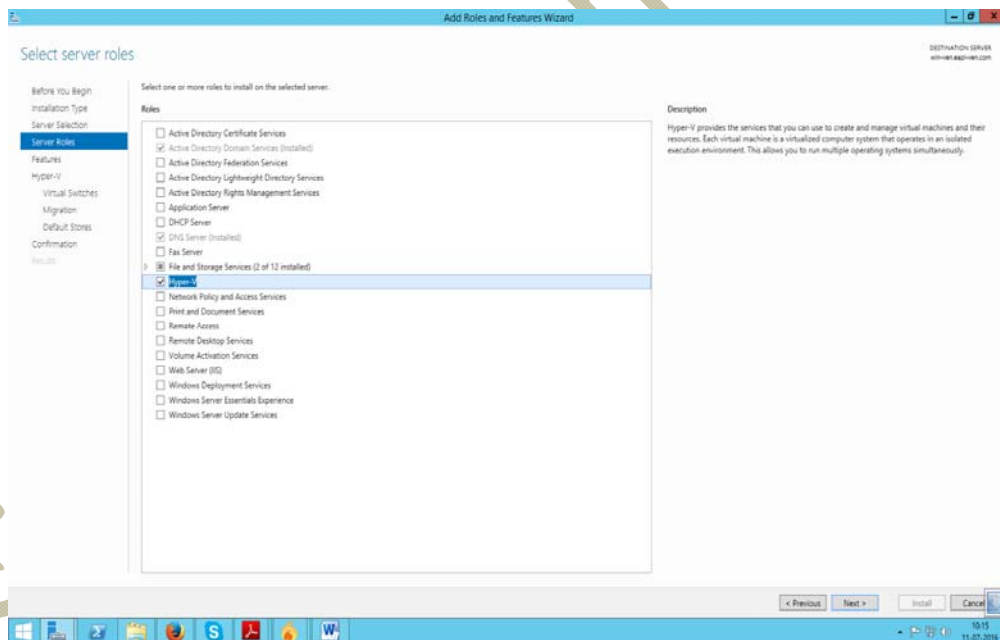
1. Go to Server Manager
2. In Server Manager, choose option 2, add Roles and Features.
3. At the Select Installation type page, choose the role-based or feature-based installation . Click Next.



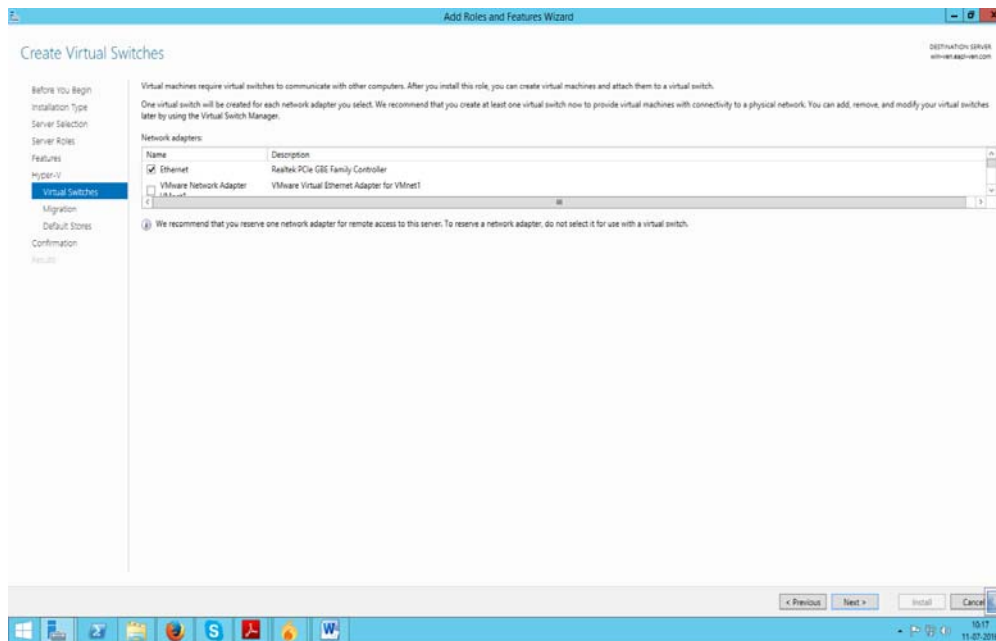
4. On the Select destination server screen. Choose select a server from the server pool and choose the server which you want to add this role. Click Next.



5. On the Select Server Roles screen, click the check box next to Hyper-V. When the Add Features dialog box appears, click the Add Features button. Then click Next.



6. At the select features screen, Click Next.
7. At the Hyper-V introduction Screen, Click Next.
8. At the Create Virtual Switches screen, chose your adapter and click next.



9. At the Virtual Machine Migration screen, click Next. You want to use migration only if you have multiple Hyper-V servers. Since we will have only one for this exercise, just skip this screen.
10. At the Default Stores screen, accept the defaults and click Next.
11. At the Confirmation screen, click the Install button.
12. After the installation is complete, click the Close button.
13. Restart your server.

Number of virtual machines utilized.

1. We can utilize n- number of virtual machines based on size of RAM and HDD

Installation of Microsoft System Center VM Manager

1. Download VHD
2. Using Hyper-V Click on Create New virtual Machine
3. Give the Machine Name
4. Give RAM as 4GB.
5. Select Install Operating system Later
6. Click the Existing Hard disk Button
7. Then Click finish.
8. It Will install the Virtual machine will install to create cloud. }

Experiment Number 2

Program to use the Map Reduce of Hadoop to interact with it

```
package mapred;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.util.*;

import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapred.*;
import org.apache.hadoop.util.*;

public class calculation {
    public static long exectime;
    static long startTime = System.currentTimeMillis();
    public static class Map extends MapReduceBase
implements
        Mapper<LongWritable, Text, Text, IntWritable> {
        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();

        public void map(LongWritable key, Text value,
            Reporter reporter)
            throws IOException {
            String line = value.toString();
            System.out.println("Line == : "+line);
            StringTokenizer tokenizer = new
StringTokenizer(line);
            while (tokenizer.hasMoreTokens()) {
                word.set(tokenizer.nextToken());
                output.collect(word, one);
            }
        }
    }
}
```

```
public static class Reduce extends MapReduceBase
implements
    Reducer<Text, IntWritable, Text, IntWritable> {
values,
    public void reduce(Text key, Iterator<IntWritable>
Reporter reporter)
        OutputCollector<Text, IntWritable> output,
        throws IOException {
            int sum = 0;
            while (values.hasNext()) {
                sum += values.next().get();
            }
            output.collect(key, new IntWritable(sum));
        }
    }
```

```
public static void main(String[] args) throws Exception {
    JobConf conf = new JobConf(calculation.class);
    conf.setJobName("wordcount");

    conf.setOutputKeyClass(Text.class);
    conf.setOutputValueClass(IntWritable.class);

    conf.setMapperClass(Map.class);
    /*conf.setCombinerClass(Reduce.class);
    conf.setReducerClass(Reduce.class);*/

    conf.setInputFormat(TextInputFormat.class);
    conf.setOutputFormat(TextOutputFormat.class);

    FileInputFormat.setInputPaths(conf, new
Path("/hadoop/mapred/system/Gaz.txt"));

    FileOutputFormat.setOutputPath(conf, new
Path("/hadoop/mapred/system/out1.txt"));
    FileSystem fs=FileSystem.get(conf);
    if(fs.exists(new
Path("/hadoop/mapred/system/JOB1.txt")))
        fs.delete(new
Path("/hadoop/mapred/system/out1.txt"));

    JobClient.runJob(conf);
}
```

```
        exectime=System.currentTimeMillis() - startTime;
        System.out.println("Completed in "+(exectime));
        /*BufferedWriter bw=new BufferedWriter(new
FileWriter("./DTG.txt",true));
        bw.write(String.valueOf(exectime));
        bw.newLine();
        bw.close();*/
        /*System.out.println("====Cluster Statu
=====");
        System.out.println(new
JobClient().getClusterStatus());*/
    }
}
```

Elysium Academy Pvt Ltd