Spam Spam Spam Spam

Calming Exim down a bit

Step 1: Prevent exim from going crazy on your container without permanently stopping it. **This is not a final solution**; this is mostly to make the below solutions easier to progress with.

VPS:

```
touch /etc/eximdisable; service exim stop ; sed -i 's/\(deliver_queue_load_max\).*/\1=4/;s/\(queue_only_load\).*/\1=4/;s/\(remote_max_parallel\).*/\1=5/;' /etc/exim.conf ; sleep 5 ; rm -f /etc/eximdisable ; service exim restart
```

Hybrid:

```
touch /etc/eximdisable; service exim stop ; sed -i 's/deliver_queue_load_max.*/deliver_queue_load_max=8/;s/queue_only_load.*/queue_only_load=8/;' /etc/exim.conf ; sleep 5 ; rm -f /etc/eximdisable ; service exim restart
```

Spam Sources

Step 2: Find your spam sources

Exploited PHP scripts:

This will find you the folders in which the source of your spam resides:

```
grep "cwd=" /var/log/exim_mainlog | awk '{for(i=1;i<=10;i++){print $i}}' |sort |uniq -c| grep cwd | sort -n | grep /home/
```

Exploited PHP Scripts (Enhanced):

Rather than just get the folders, you might be able to get **the actual PHP script files that are responsible**!

First, install GNU parallel:

```
rpm -i http://updates-vps.wiredtree.com/centos/4/wt-extra//noarch/wt-parallel-20141122-1.noarch.rpm
```

Second, you'll want to use these two lines. They'll provide you the location of every script being actively used to send spam, via the access logs:

```
export spamsources=$(cat /var/log/exim_mainlog | grep 'cwd=.*public_html' | awk
'{print $3}' | sort | uniq -c | sort -nr | egrep [0-9]{4}' ') ; echo "$spamsources"
export spamscripts=$(echo "$spamsources" | parallel -k "bash <(curl -s -L
files.wiredtree.com/spam/php-sources.sh)" | grep ^.home | sort | uniq | grep -v
'public_html.wp-cron.php\|\/$') ; echo "$spamscripts"
```

If you get results from the above script, and **those files exist**, archive all the scripts into a date-stamped file in /root/abuse:

```
export spamarchive=$(echo "spam".$(date +%Y.%m.%d)".tar.gz") ; echo $spamarchive;
mkdir -p /root/abuse/
echo "$spamscripts" | tar zcvf /root/abuse/$spamarchive -T - --remove-files
```

Finally, purge the queue of the scripts' e-mails:

```
echo "Failure messages:" ; exiqgrep -i -f '<>' | xargs exim -Mrm | wc -l
echo "Total script spam messages:" ; exim -bp | grep \< | grep -f <(tar tf
/root/abuse/$spamarchive  | awk -F '/' '{print $2}' | sort | uniq)  | awk '{print $3}'
| grep -v \< | xargs exim -Mrm | wc -l
```

Responsible outbound e-mail address

(warning: can be faked):

```
cat /var/log/exim_mainlog | grep " <= " | awk -F ' <= ' '{print $2}' | awk '{print$1}'
| sort | uniq -c | sort -nr | head
```

Exploited Logins

(e.g. e-mail passwords):

```
cat /var/log/exim_mainlog | grep 'dovecot_login' | awk -F 'dovecot_login:' '{print
$2}' | awk -F ' ' '{print $1}' | grep -v '^$' | sort | uniq -c | sort -nr | head -n 40
```

Additional Reading:

You'll note that all of the above entries start like this:

```
cat /var/log/exim_mainlog
```

This is not a mistake, or deliberate misuse of `cat`. This is actually purposeful. Under some circumstances, you may need to make modifications to that line, and it helps to have something "modular" that you can easily update.

For instance, let's say you want to go through **all** of the mail logs? Try replacing that chunk of code with this:

```
cat /var/log/exim_mainlog <(find /var/log/exim_mainlog*gz -exec gunzip -c {} \;)
```

Or let's say you only want to check the last few thousand log lines? Try replacing it with this:

```
tail -n 5000 /var/log/exim_mainlog
```

Or maybe you only wanna search through logs from June 12th, 2015? Try this:

```
cat /var/log/exim_mainlog | grep 2015.06.12
```

Purge the Spam Queue

First, you'll want to clear the queue of all failure notices:

```
exiqgrep -i -f '<>' | xargs exim -Mrm | wc -l
```

From there, **once you remove the spam-creating scripts from the server,** you can remove the e-mails themselves like this:

("example_user" in this case is the cPanel account that was hosting the spamming files)

```
exim -bp | grep '<.*example_user\|<>' | awk '{print $3}' | grep -v \< | xargs exim
-Mrm | wc -l
```